



Presented by Sally Cox



# HTML5 for Beginners

# LEARN BY VIDEO

**WORKSHOP**



Smartphone Users:  
Scan for a Video Sample





Presented by Sally Cox

# HTML5 for Beginners

# LEARN BY VIDEO

WORKSHOP



## **HTML5 for Beginners: Learn by Video**

**video2brain GmbH • Sally Cox**

Copyright © 2012 by video2brain GmbH

Peachpit Press

1249 Eighth Street

Berkeley, CA 94710

510/524-2178 • 510/524-2221 (fax)

Find us on the Web at [www.peachpit.com](http://www.peachpit.com)

To report errors, please send a note to [errata@peachpit.com](mailto:errata@peachpit.com)

Peachpit Press is a division of Pearson Education.

Project Editor: Tracey Croom

Media Producer: Eric Geoffroy

Proofreader: Liz Welch

Package design: Charlene Charles-Will

Book interior design: Danielle Foster

### **Notice of Rights**

All rights reserved. No part of this video training book may be reproduced or transmitted in any form by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher. The software and media files on this disc are copyrighted by the authors and Peachpit. You have the non-exclusive right to use these programs and files. You may use them on one computer at a time. You may not transfer the files from one computer to another over a network. You may transfer the files onto a single hard disk so long as you can prove ownership of the original disc. For information on getting permission for reprints and excerpts, contact [permissions@peachpit.com](mailto:permissions@peachpit.com).

### **Notice of Liability**

The information in this video training book is distributed on an "As Is" basis, without warranty. While every precaution has been taken in the preparation of the video training book, neither the trainers nor video2brain GmbH nor Peachpit shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the instructions contained in this video training book or by the computer software and hardware products described in it.

### **Trademarks**

All rights reserved. Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this video training book, and Peachpit was aware of a trademark claim, the designations appear as requested by the owner of the trademark. All other product names and services identified throughout this video training book are used in editorial fashion only and for the benefit of such companies with no intention of infringement of the trademark. No such use, or the use of any trade name, is intended to convey endorsement or other affiliation with this video training book.

ISBN-13: 978-0-321-81542-2

ISBN-10: 0-321-81542-4

9 8 7 6 5 4 3 2 1

Printed and bound in the United States of America

# DVD Contents

## 1 What Is HTML5?

---

- 1.1 Understanding Hypertext Markup Language (HTML)
- 1.2 Learning the Parts of a Web Page
- 1.3 Writing Your First Snippets of Code
- 1.4 Understanding How Browsers Interpret HTML
- 1.5 Comparing Web Browsers and How They Display Web Pages

## 2 Understanding HTML5 Markup and CSS3

---

- 2.1 Examining HTML Tags
- 2.2 Incorporating HTML5 Tags
- 2.3 Adding Style with CSS3
- 2.4 Working with CSS and HTML5 Together

## 3 Beginning with Wireframing

---

- 3.1 Deciding How Your Site Will Work
- 3.2 Planning the Hierarchy of Your Site
- 3.3 Creating Your First Wireframe
- 3.4 Thinking About Usability

## 4 Creating a Simple Blog Design with HTML5

---

- 4.1 Changing the DOCTYPE of Your HTML5 Document
- 4.2 Understanding the Elements of a Blog
- 4.3 Adding a Header and Footer to Your Blog
- 4.4 Posting to Your Blog
- 4.5 Adding a Sidebar with the Aside Element
- 4.6 Using Lists to Organize Content
- 4.7 Creating Tables in a Web Page

## 5 Designing an HTML5 Form

---

- 5.1 Beginning to Design a Form
- 5.2 Understanding Basic Form Elements
- 5.3 Adding an Attribute to the New HTML5 Input Tag
- 5.4 Highlighting Important Content

## 6 Incorporating Graphics

---

- 6.1 Understanding Web Graphic Formats
- 6.2 Saving Graphics Properly for the Web
- 6.3 Working with Color on the Web
- 6.4 Resources for Designing with Color
- 6.5 Exploring the Canvas Tag

# DVD Contents

## 7 Adding Multimedia

---

- 7.1 Why Use Multimedia?
- 7.2 Adding Audio to Your Contact Form
- 7.3 Understanding Video Types for the Web
- 7.4 Working with Video Controls

## 8 Conclusion

---

- 8.1 Updating Your Blog
- 8.2 Changing the Look and Feel of Your Site
- 8.3 Conclusion



# Booklet Contents

## Introduction vii

---

## Common HTML5 Tags and What They Do 1

---

- Looking into the Future . . . . . 2
- New Multimedia Tags . . . . . 5
- New Web Application/Ajax/  
DHTML Tags . . . . . 6

## Fonts and HTML5 9

---

- A Brief History . . . . . 10
- Different Strokes for Different Folks . . . 13
- Accessing the Google Fonts Library . . . 18
- A Beginner's Primer on Designing  
with Fonts on the Web. . . . . 21
- Creating Accessible Web Design  
and Typography . . . . . 24

## Glossary 27

---

# Introduction



# Introduction

## Welcome to *HTML5 for Beginners: Learn by Video*

The advent of HTML5 has opened up new frontiers in the world of web design, and in this course you'll learn how to exploit the power of HTML5 to make the best-looking, smoothest-functioning, and easiest-to-use web pages possible.

*HTML5 for Beginners: Learn by Video* begins with an overview of HTML and CSS so you'll feel more comfortable working with the code. Then you'll learn how to create an elegant blog site with HTML5, starting with the concept and proceeding to build a wireframe that serves as the foundation of the site. We'll build on that wireframe to create a professional-looking blog site, complete with header and nav bar, and even add an interactive form. Finally, we'll add graphics, color, and multimedia to give our content visual appeal.

Along the way you'll learn about the new tags in HTML5, such as the Video tag, which allows you to embed video directly within the web page instead of pointing the user to a server. You'll also get lots of tips on web design, such as which graphic formats to use and the best way to save your graphics for the Web.

These videos are designed for easy navigation, and you can even bookmark your favorite sections to review again and again. You can also test yourself on what you've learned with multiple-choice quizzes. We hope you find this video training and booklet useful and informative and that you are excited about the websites you create with your newly acquired skills.

## About This Book

While the video training gives you an overview of HTML5 and allows you to experiment with some of the more popular HTML5 tags, this booklet will give you a complete list of HTML5's new tags and help you decipher which browsers will work with the new tags. (Not all tags are supported in all the major browsers, so you won't want to incorporate all HTML5 tags at this time.) We'll also discuss some options for getting the fonts you choose to display as intended on your end user's screen—such as the CSS rule `@font-face` and the Google Fonts Library—and some best practices for designing with type on the Web.

## How to Use This Course

The video2brain interface is easy to use, but it also has a large number of options. We've provided this brief guide to give you a tour of the interface and make sure you don't miss out on any of its features.

This course comes in a few different flavors. You can run the Mac application (labeled "Start" or "Start.app") or the Windows application (.exe) file, or you can open `start.html` in your browser. Most people will use one of the application versions of the course. If you're working with an operating system that doesn't support one of these two options, such as Linux, the browser version is a great alternative—it works on any browser with a current version of Adobe Flash. The main difference between running the course as an application versus opening it in your browser is that the application versions include an option, discussed

in more detail later in this guide, called Watch-and-Work mode. Watch-and-Work mode lets you watch the course in a smaller window while you follow along in another application. This option doesn't exist in the browser version of the course.

Whether you're using the application or browser version, when you first launch the course, you will see the Welcome screen.



## The Welcome Screen

From here, there are a few ways you can proceed:

- Click the Play icon in the middle of the screen to watch a video that introduces you to the instructor(s) and gives you an overview of what's covered in the course.
- For onscreen help and a user's manual, click either the Help item on the menu at the top of your screen or the Need Help item at the bottom of the Content panel. You can also access help by pressing F1 on your keyboard.

- Entering a word or phrase in the Search field in the upper right of the screen will search the course's titles and descriptive text to help you find what you're looking for.

## Starting the Course

When you're ready to dive in, you have several options for how to get started:

- To begin at the beginning and play all the way through, click Play All Videos.
- To access a list of every video available in the training, click Table of Contents.
- To start with Chapter 1 (or any chapter that appeals to you), click the chapter name on the left of your screen.

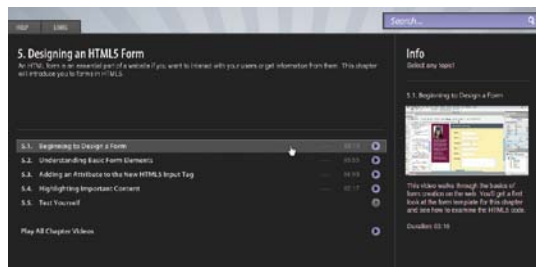
1. What Is HTML5?
2. Understanding HTML5 Markup and CSS3
3. Beginning with Wireframing
4. Creating a Simple Blog Design with HTML5
5. Designing an HTML5 Form
6. Incorporating Graphics
7. Importing Multimedia into Your Blog
8. Conclusion

You can navigate the Table of Contents and Chapter views using your mouse, or using the keyboard to arrow up and down, left and right. When you've selected a movie, you can press the right arrow to play that movie.

# Introduction



In the Table of Contents or Chapter view, any time you move your mouse over the name of a chapter or video, text describing the contents of that item will appear on the right side of your screen. This area is known as the Sidebar and is discussed in detail on the next page.



If you click the name of a chapter, you'll see a list of all the videos in that chapter. Choose the video you want to start with or click Play All Chapter Videos to start with the first video in the chapter and go all the way through.

## KEYBOARD SHORTCUTS FOR THE PLAYER

- **Spacebar:** Play/pause
- **Right Arrow:** Jump forward
- **Left Arrow:** Jump backward
- **Tab:** Show/hide Sidebar
- **B:** Add bookmark
- **M:** Show/hide navigation bar
- **Esc:** Turn off Full-Screen mode

## Navigating Inside a Video

While watching a video, you can use the buttons along the bottom right of your screen to toggle both video smoothing **1** and Full-Screen mode **2** on and off. You can adjust the volume **3** from here as well. You can also click and drag the green line **4** in the timeline at the bottom of the screen to move around within the video.

When the video ends, click the Next Video button at the top right of the window to go directly to the next lesson. (This will happen automatically if you clicked Play All Chapter Videos to launch the chapter.) At the end of a chapter, this option will change to Next Chapter. You can also click Training Content at any time to leave the player and choose another lesson or click the navigation bar near the top of the screen to access the table of contents.





## Resizing Your Window

The player interface gives you a few different options for resizing your window to fit your needs.

Click the Full-Screen icon **1** to enter Full-Screen mode. You can exit Full-Screen mode by clicking the icon again or by pressing the Esc key. You can also dynamically resize your window while watching simply by grabbing the lower-right corner of the window and dragging until the window is the size you want.

In the application versions of the course, you can also choose Watch-and-Work mode **2** from the player. This takes you to a smaller screen that leaves room for you to work in another application alongside the course. Click the Standard button to return to a regular-size screen.

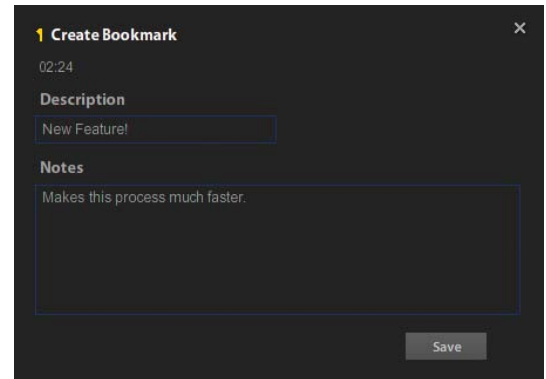
## The Sidebar

The Sidebar is an area on the right side of the player where additional information, such as a description of the video you're watching, is displayed. At the bottom are buttons that enable you to access your bookmarks or a list of videos in the chapter. To turn the Sidebar on or off, you can press Tab or click the button in the navigation bar.



## Using Bookmarks

It's easy to create a bookmark in the video to mark where you left off or make note of something you want to refer back to later. Just click the Bookmark button at the bottom right or use the B keyboard shortcut. You will see the Create Bookmark dialog box.

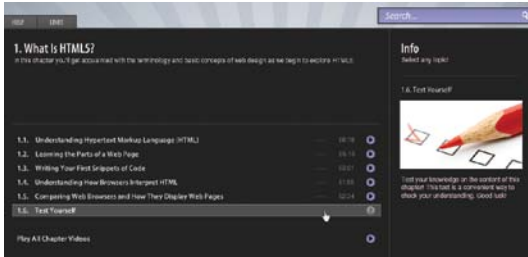


Enter a name for your bookmark and an optional note, then click Save or press Enter/Return. The bookmark will then be visible as a thin line in the timeline. You can access your bookmarks by clicking My Bookmarks at the top of the screen or via Video Bookmarks in the Sidebar. Just click the bookmark to go to your previously marked point. To delete a bookmark, click the Trashcan icon to the right of the bookmark and confirm your deletion by clicking Yes.

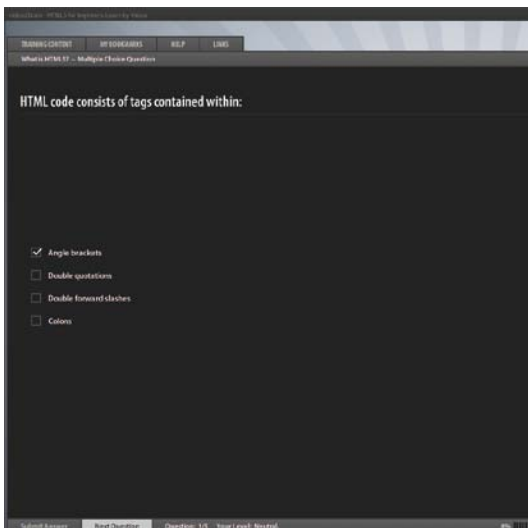
# Introduction

## Test Yourself

This course includes a number of Test Yourself sections, each of which contains a series of questions about the topics covered in that chapter.



Just click the box for the right answer. Some questions refer to an image that is displayed on your screen.



After answering any question, click the Submit Answer button at the bottom left to indicate that you're finished. A dialog box will pop up to let you know whether you got the question right. (If you get a question wrong and want to see the correct answer, press and hold **F**, **A**, and **N** on your keyboard at the same time. A check mark will appear in the box next to the right answer.) This dialog box also contains a button that will take you to the next question.

When you're finished answering all the questions, click Training Content to return to the table of contents.

# About the Authors

## About video2brain

video2brain (video2brain.com) has been Europe's premier source for video training since 2002. We produce high-quality English, German, French, and Spanish video training on a variety of software topics, with a special emphasis on graphics, web design, photography, and programming. Our customers include people just starting out, professionals, educational institutions, and global corporations as well as home enthusiasts.

All of our courses are available in 1280 x 720 Hi-Def video, with a full-screen mode that creates an immersive learning experience. They also include an interactive, easy-to-use interface; custom bookmarks that let you annotate your course and remember where you left off; and "Watch-and-Work" mode, so you can practice as you learn.

We're passionate about teaching and about helping you build exceptional skills so you can create extraordinary work. Our mission is to provide the best and most affordable video training possible, and to offer you a feature-rich learning environment aimed at enhancing the effectiveness of our training.

video2brain was founded by Gerhard Koren, a v2b trainer himself. Our home office is nestled among the mountains of the Austrian Alps in the city of Graz, a "student city" with six universities.

## About Sally Cox



Sally Cox has 12 years of experience in the printing/publishing industry and is a former art director for Adobe Systems on the multi-media team. She is an Adobe Certified Software Trainer and

published author of books and white papers on Adobe products. Her work includes instructional design and creating eLearning content for [kreatable.com](http://kreatable.com), using Adobe Captivate, Connect Pro, and Flash. Sally has taught graphic design courses at two Art Institute of California campuses and continues teaching Adobe software, both in the classroom and online via Adobe Connect Pro. She delights in combining her print background with new technologies to stay on the cutting edge.



# Common HTML5 Tags and What They Do



# Looking into the Future

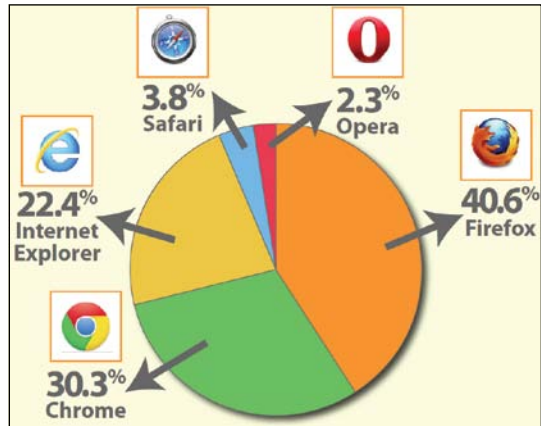
---

With the advent of HTML5, web design has taken a bold new step forward by integrating a variety of technologies, including HTML, CSS, and JavaScript. Some older tags have been eliminated and new ones have been added. Let's take a look at the new HTML5 tags and get a feel for their purpose.

## Browser Support

Along with a description of each element, I will include a note about what browsers support it and, where applicable, a list of possible attributes to apply to the element.

The top five browsers in the marketplace today are, in order of usage, Firefox, Chrome, Internet Explorer, Safari, and Opera.



## New Layout Tags

HTML5's new layout tags allow you to define sections of your blog or web page as "layout regions" without having to use a DIV tag to define them. Your site's header is encompassed within the header tag, and your navigation or "nav bar" is surrounded by the <nav> tag. You simply use these tags just like you would a DIV tag for your layout—only now they are named for you, so you don't need an ID unless you want one.

### Layout Tags in HTML5

```
<article>
<aside>
<figure>
<footer>
<header>
```

## Article

The Article tag is used for the main content of your blog or website. Examples of relevant usage of the Article tag include newspaper articles, forum posts, user comments, and of course blog entries. It is supported by all five major browsers.



## Aside

The Aside tag is used to create a “sidebar” of content, usually content that is relevant to the adjacent content. It is supported by all five major browsers.

In this example, the burgundy-shaded area on the left was created with the Aside tag.



## Figure

The Figure tag is used to specify content that is contained within the flow of the main content, such as images, diagrams, code, and photos. It is contained within the actual content flow, not outside it, so its removal should not affect the document's flow. It is supported by all five major browsers.

In the mock-up example shown, the image is within the text content, so a Figure tag is appropriate in this situation.



## Footer

The Footer tag denotes content to be placed at the bottom of the page and eliminates the need for a DIV tag to contain this content. It is supported by all five major browsers. To use it, simply place your footer content inside the Footer tag in HTML5.



## Header

All content to be included in the header of your blog or web page is now included in the new Header tag. It is supported by all five major browsers.



# New Multimedia Tags

The new multimedia tags in HTML5—Audio, Source, and Video—make a valiant effort to simplify the process of adding audio and video to web pages. Although they are not perfect, they are a significant step in the right direction.

## Audio

Music or other types of audio streams are now added to web pages via the Audio tag. It is supported by all five major browsers.

Attributes for the Audio tag include Autoplay, Controls, Loop, Preload, and SRC.

```
<audio controls="controls">
<source src="covered_bridge.ogg"
  type="audio/ogg" />
<source src="covered_bridge.mp3"
  type="audio/mp3" />
Your browser does not support
the audio element.
</audio>
```

## Source

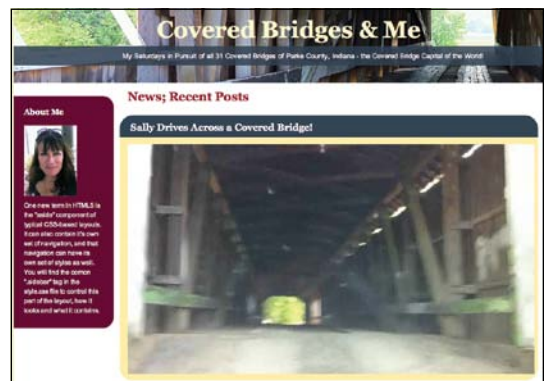
The Source tag identifies media resources required for media elements like the Video and Audio tags. It is supported by all five major browsers. Attributes for the Source tag include Media, SRC, and Type.

```
<audio controls="controls">
  <source src="song.ogg" type="audio/ogg" />
  <source src="bridge.mp3" type="audio/mpeg" />
Your browser does not support the audio element.
</audio>
```

## Video

The Video tag has caused much excitement among developers. One key advantage to the Video tag is the ability to embed multiple formats, so the browser can make the decision to play the best choice.

Shown here is an example of a web page with a video using the Video tag. Attributes for the Video tag include Audio, Autoplay, Controls, Height, Loop, Poster, Preload, SRC, and Width.



# New Web Application/Ajax/DHTML Tags

---

Advanced tags include those designed specifically for web applications, Ajax, and DHTML: Canvas, Command, Datagrid, Datalist, Details, and Output.

## Canvas

The Canvas tag creates a container to hold graphics. It works in conjunction with CSS and JavaScript.

Attributes of the Canvas tag include Height and Width.

```
<canvas id="SallyCanvas"></canvas>
<script type="text/javascript">
var canvas=document.getElementById("SallyCanvas");
var ctx=canvas.getContext('2d');
ctx.fillStyle='#FF3300';
ctx.fillRect(0,0,86,100);
</script>
```

## Command

The Command tag is only supported in Safari at this time and is used to define a command button. Examples include a radio button, checkbox, or button.

```
<menu>
<command onclick="alert('Covered Bridge')">
Click Me!</command>
</menu>
```

## Datagrid

The Datagrid tag creates an interactive representation of tree, list, or tabular data. It can also be used to provide access to tabular data via a scripting language.

## Datalist

At this time, the Datalist tag is only supported in Opera and Firefox, and is used in conjunction with the Input element to define which values the Input element can have.

```
<input list="CoveredBridges" />
<datalist id="CoveredBridges">
<option value="Britton">
<option value="Coxe Ford">
<option value="Bridgton">
</datalist>
```

## Output

The Output tag is used to display the results of a calculation. This is typically done within a form. Attributes for Output include For (used to explain the relationship between the calculation and the result), Form (indicates which form the output element belongs to), and Name (the unique name of the output element).

## Conclusion

These are the main HTML5 tags at this time. Others include Progress (a progress bar), Dialog (a conversation), Hgroup (a grouping of header tags that needs to remain intact), Mark (marked up or highlighted text), Meter (used for measurement purposes), and Time (indicating a date or time).



# Fonts and HTML5



# A Brief History

---

For as long as the World Wide Web has been in existence, fonts have been a weak link in the web design chain. Web developers could specify a particular font for a particular section or page, but if the client didn't have that font installed on their local computer/system, the text would not appear as intended.

Traditionally, developers needed to set up a substitute font list (i.e., Arial, Verdana, Helvetica) so that when the intended font was not present, an appropriate alternate could be substituted. However, the end user could set up their web browser to use certain fonts on the list, taking away even more control over the look and feel of the pages from the developer. With the addition of CSS3 to the Web, we finally have an answer to this problem: downloadable fonts!

## @ font-face

@font-face is a CSS rule that enables the designer to assign a particular font to text and, if the end user doesn't have that font installed, have the font downloaded from the server to render the text as intended. This frees web designers from using the short list of "web-safe" fonts that come preinstalled on most computers.

In CSS2, @font-face was presented to the world, relying on the Embedded Open Type (.eot) format. Unfortunately, the only browser that supported it was Internet Explorer and it really didn't go anywhere at that time. Then Safari 3.1 released, and suddenly, we were able to use TrueType (TTF) or OpenType (OTF) fonts on our pages.

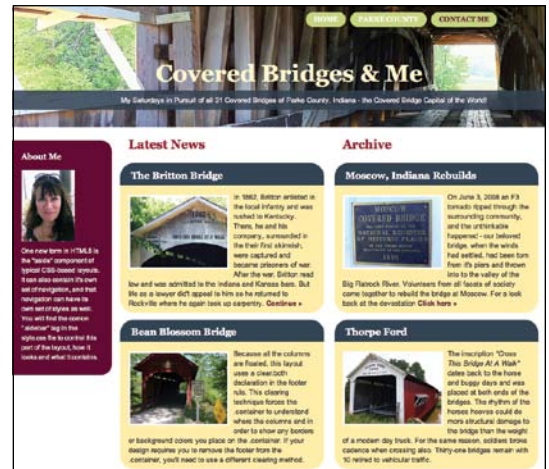
As of this writing, the latest versions of Safari, Firefox, and Google Chrome support @font-face, and it is expected that Opera will support it soon.

## How to Use @font-face

To use @font-face, you begin by defining the CSS rule. font-family is the name of the font, src is where it is located, and you can optionally include a font-weight. A font-weight is not needed for regular type, but should be included for other styles like bold, thin, etc.

When you set up the @font-face element, each component of the font family is indicated using the @font-face rule. For instance, to use both regular and italic forms of the Eurostile font, you would set up your style sheet as indicated in the graphic shown here.

When you are ready to use this font, you call for it using the font-family element, as indicated above. Since you already preset it in the CSS file, the browser knows how to display it.



```
@font-face { font-family: Eurostile; src: url('Eurostile-Roman.otf'); } @font-face { font-family: Eurostile; font-weight: bold; src: url('Eurostile-Bold.otf'); }
```

```
h3 { font-family: Eurostile, sans-serif; }
```

Eurostile Regular looks like this and will be rendered in Safari 3.1 as shown, and **Eurostile Bold** looks like this.

## The Future of @font-face

The goal in the web community is to have an online repository of free fonts that work with the @font-face CSS rule. There are restrictions right now, due to licensing issues of certain fonts.

The plan is this: font designers/foundries would upload their fonts and have certain rights by doing so. They would own the font outright, but would allow it to be used with @font-face for free. They would also allow designers/developers to link to it from @font-face.com.

This would still give the type houses ownership of the fonts, but would give the fonts exposure and recognition. It would free designers and developers from the drudgery of either using web-safe fonts or creating graphics from their designer fonts.

At this time, not every type house is on board with this idea. Understandably, they are concerned about protecting their licensed products. It is the hope of the web community that all fonts will someday be available for use with @font-face in this manner.

## References for @font-face

Here is a short list of URLs for helpful sites where you can learn more about @font-face:

- <http://www.fontsquirrel.com/fontface/generator>
- <http://www.ironpaper.com/webintel/articles/html-5-font-face-font-format-usage/#font-face-reference>
- <http://www.css3.info/preview/web-fonts-with-font-face/>

# Different Strokes for Different Folks

Different browsers handle fonts differently. As previously mentioned, Internet Explorer requires an .eot font and the other browsers must have a .ttf (TrueType) or .otf (OpenType) font.

When a web font is loading in Safari, Chrome, or Internet Explorer, a blank spot appears where the text will display. Opera and Firefox, however, show the text to be styled in a default font, which then is replaced by the correct font when it is fully loaded. The acronym for this is FOUT (Flash of Unstyled Text), and it is frowned upon by many in the design community.

## WebFont Loader

Having the type display differently in different browsers, and not being able to control it as the web font is loading, may seem problematic. Fortunately, there is a way around this! The WebFont Loader, a JavaScript library co-developed by Google and Typekit, can make all the browsers perform the same way while the web font is loading. It offers users more control over font loading than the Google Web Fonts API provides and allows you use multiple web-font providers. It does, however, require a basic knowledge of JavaScript.

If this all seems too overwhelming to you, remember that you can always hire someone to help with the JavaScript part of your web or blog site.



```
<html>
<head>
<script type="text/javascript">
  WebFontConfig = {
    google: { families: [ 'Tangerine', 'Cantarell' ] }
  };
  (function() {
    var wf = document.createElement('script');
    wf.src = ('https:' == document.location.protocol ? 'https' :
'http') +
    '://ajax.googleapis.com/ajax/libs/webfont/1/webfont.js';
    wf.type = 'text/javascript';
    wf.async = 'true';
    var s = document.getElementsByTagName('script')[0];
    s.parentNode.insertBefore(wf, s);
  })();
</script>
<style type="text/css">
.wf-loading p {
  font-family: serif
}
.wf-inactive p {
  font-family: serif
}
.wf-active p {
  font-family: 'Tangerine', serif
}
.wf-loading p {
  font-family: serif;
  font-size: 16px
}
a {
  font-family: serif;
  font-size: 16px
}
.wf-active h1 {
  font-family: 'Cantarell', serif;
  font-size: 16px
}
</style>
</head>
<body>
<h1>This is using Cantarell</h1>
<p>This is using Tangerine!</p>
</body>
</html>
```

## WebFont Loader Resources

Here are a few URLs for sites where you can learn more about the WebFont Loader and web font hosting services:

- [http://code.google.com/apis/webfonts/docs/webfont\\_loader.html](http://code.google.com/apis/webfonts/docs/webfont_loader.html)
- <http://24ways.org/2010/using-the-webfont-loader-to-make-browsers-behave-the-same>
- <http://sprungmarker.de/wp-content/uploads/webfont-services/>

## Adobe Typekit

As this text went to print, it was announced that Adobe Systems, Inc. had purchased Typekit. This was big news in the Adobe community, as Typekit is a well-respected web fonts service that offers hundreds of standards-compliant and legal web fonts. Check out their website, [typekit.com](http://typekit.com), for a feature tour and to get a free trial of their quality web font service.

## Google Fonts

Google Fonts give web designers and developers access to the Google Font Library so they can use fonts whether or not those fonts are installed on the end user's computer. They simply link their documents to one or more fonts in the Google Font Library, and those fonts are displayed as expected when the document is viewed online.



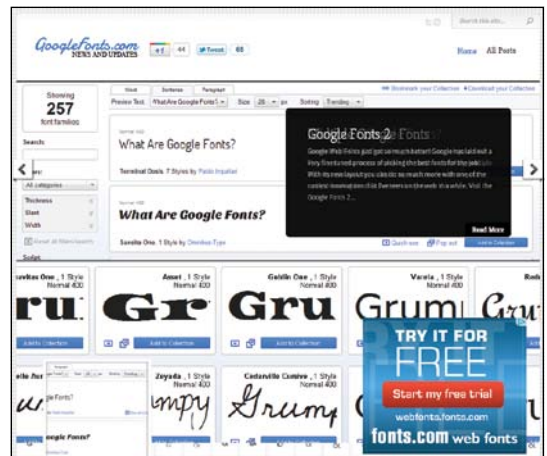
You can use multiple Google Fonts in the same document. If you are accessing multiple fonts within the same font family, Google makes it easy by using a piece of code that accesses the entire family at once. This avoids the extra loading time that might occur if the browser had to load each font individually.

The code shown here helps speed up the process of designating the font family in a web page.

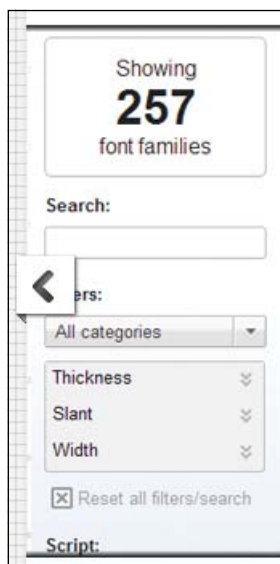
```
http://fonts.googleapis.com/css?family=Google Fonts  
Name 1 | Google Fonts Name 2 | Google Fonts Name 3
```

## The New and Improved Google Fonts Site

Googlefonts.com has a completely revamped site, which gives you clean and refined views of all the popular Google Fonts. On the left side of the screen, you'll see how many fonts are available at the time you are visiting the site.



Just under that listing, there is a search box, in case you know the name of the font you want to find. Under the Search field is the Filters section. This lets you home in on the characteristics of the font(s) you are searching for, i.e., sans serif or ultra-light.



## Getting Started with Google Fonts

There are many options for fonts that work in conjunction with HTML5, and Google Fonts is a popular choice. But what if you don't know where to begin? Let's take a closer look at how to use Google Fonts.

To use Google Fonts in your website, you simply link to an external style sheet, and then refer to a font in that style sheet on your web page via a CSS style.

Here is an example of the code you add to the head section of your HTML5 document to access the Google Fonts style sheet.

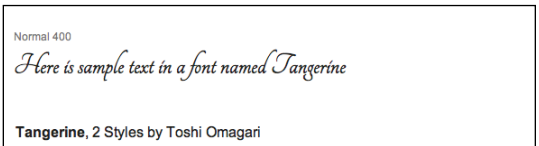
```
<html>
<head>
  <link rel="stylesheet" type="text/css" href="http://fonts.
googleapis.com/css?family=Tangerine">
  <style>
    body {
      font-family: 'Tangerine', serif;
      font-size: 48px;
    }
  </style>
</head>
<body>
  <h1>Making the Web Beautiful!</h1>
</body>
</html>
```

When you open that web page in a browser, the text will appear in the correct font—in this case Tangerine. You can even apply formatting, such as a drop shadow, to Google Fonts. This opens up a wide range of design possibilities.

To get started, follow this process:

1. Add a style sheet link in the head section of the HTML5 document to tell the browser where to access the fonts.
2. Style an element with the desired web font. You can do this in one of two ways. You can use a style sheet to call for a particular font, as shown in the first example here. Or you can use an inline style, which is added precisely where you want the font to appear, as shown in the second example.

It's recommended that you include a web-safe font as a backup for your Google font. Adding "serif" or "sans serif" at the end of the list can give you some control over the defaulted font.



```
<link rel="stylesheet" type="text/css" href="http://fonts.
googleapis.com/css?family=Font+Name">
```

```
CSS selector {
  font-family: 'Font Name', serif;
}
```

```
<div style="font-family: 'Font Name', serif;">Your text</div>
```



# Accessing the Google Fonts Library

To see the hundreds of free, open source Google Fonts available for your use, go to the Google Fonts Library link.

You can create your own custom collections of fonts from the library. For example, if you have several clients who like particular fonts, you can create custom sets to use for each client. Or you can group them by category, such as “script” or “display.”

<http://www.google.com/webfonts#HomePlace:home>

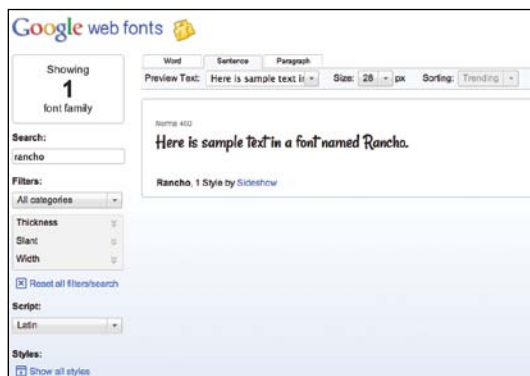
## Choosing Google Fonts

Once you have accessed the Google Fonts Library home page, you begin by selecting fonts.

Click “Start choosing fonts” to begin accessing the vast library of free Google Fonts. Next, you can peruse the list of fonts or search by name. Click in the Search field on the left side of the screen to search by name.

If you want to see a particular line of type in that font, enter the sample text in the Preview Text area at the top of the screen. You can choose a size here as well.

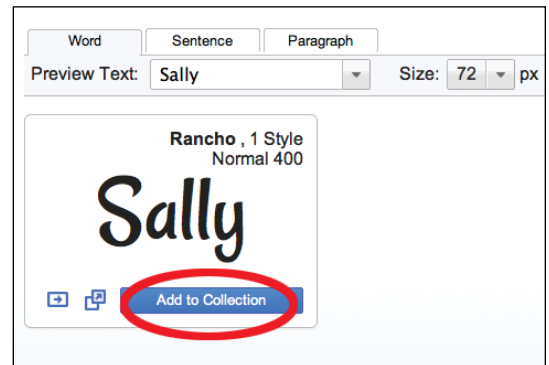
The Filter drop-down menu helps you pare down your choices. For example, if you are looking for just sans serif fonts, or handwritten fonts, you can indicate that here.



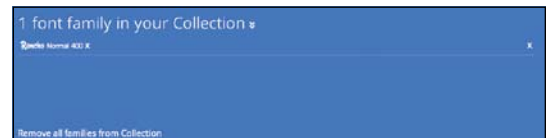
The Word, Sentence, and Paragraph tabs near the top of the screen help you get a clearer vision of what your end result will be. If you are just choosing the font for a single word, there is no sense in viewing an entire sentence, so click the Word tab and view a single word. Again, if you want to preview a particular word, enter it in the Preview Text area. You can also use the Paragraph tab to view an entire paragraph as it will appear in the selected font.



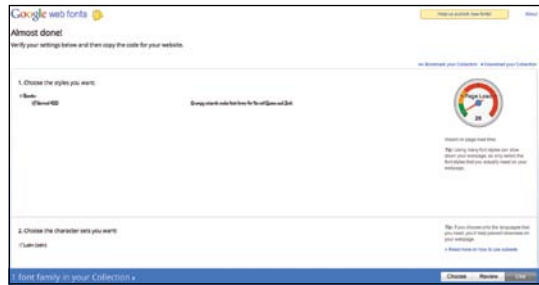
When you are happy with the font and want to add it to your collection, click the Add to Collection button next to the preview. This will allow Google Fonts to remember your choice so you can access that font again quickly and easily.



A listing at the bottom of the screen tells you how many fonts are in your collection. You can also choose to remove fonts from the collection if you change your mind by clicking the "X" in your collection listing.



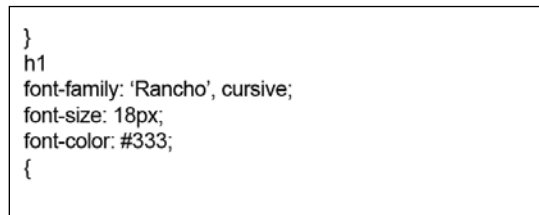
When you are ready to use the font, click the Use button at the lower-right corner of the screen. The next screen you see will allow you to copy the code you need to use that font.



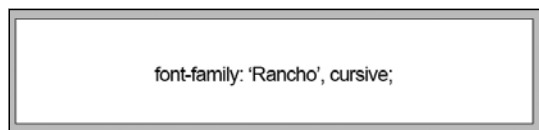
Scroll down on the page to see the code you need to add to your website. Copy this code and paste the link as the first element inside the head section of your document.



Now, you simply add the font(s) to your style sheet styles and the Google Fonts Library will access and display the fonts as needed.



The Google Web Fonts API will automatically generate the appropriate browser-specific CSS to allow you to use the fonts. You simply add the font name to your CSS styles. The example at the right shows how it works.



That's all there is to it!

The possibilities of using fonts on the Web now are endless. No more saving type as a graphic or worrying about whether the end user has the right fonts installed. Although there are other options for using fonts in conjunction with HTML5, Google Fonts are a strong choice.

There are other more advanced techniques available on the Google Fonts website. I urge you to explore the world of Google Fonts, as they are free and give both the designer and developer lots of options to play with.

# A Beginner's Primer on Designing with Fonts on the Web

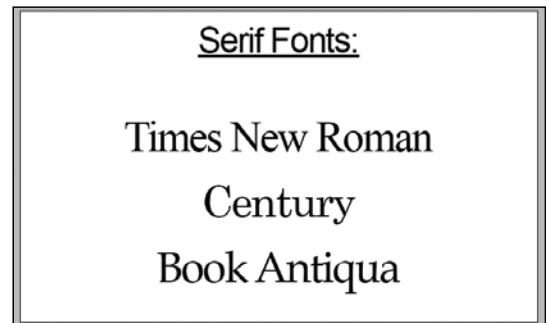
---

If you are new to type and web design, there are a few things you should know before proceeding. Although I won't claim to make you an expert after this short reading, I do hope to introduce you to some common terms and give you some things to consider when choosing fonts for your site.

## Serif Fonts

Serif fonts have little tails (or serifs) at the end of the character strokes, which help make them more readable. The basic theory is that the tails lead the reader's eye from one character to the next, increasing reading speed. You can take a tip from books and newspapers, as most of them use serif fonts for heavy text areas.

Examples of serif fonts include Times New Roman, Century, and Book Antiqua.



## Sans Serif Fonts

Sans serif fonts do not have the tails on the letters (the word "sans" means "without" in French). They can still be quite readable but are generally considered more modern in look and feel. It is no accident that the sans serif font Helvetica is used so heavily throughout the world in logos, in headlines, on billboards, and so on; it is a very readable and elegant font. Sans serif fonts are not recommended, however, for long passages of text. It is thought that the letters will start to blur and reading speed will slow down over time.

Examples of sans serif fonts include Helvetica, Arial, and Verdana.



## Display Fonts

Display fonts are flashy fonts that are generally not considered a good choice for anything other than a simple line of type. They can be good choices for eye-catching headlines, but most are difficult to read when used for more than a few words. I suggest not adding stylistics like drop-shadows to display fonts if you can help it; this just makes them that much harder to read.

Examples of display fonts include Frankfurter, Chalkboard, and Marker.

## Adding Color to Your Type

Here are a few tips for working with color in conjunction with text:

- As a general rule, black is the best choice for text on the Web, particularly for long passages. However, a dark color can be an equally compelling choice.
- Type is most readable against a lighter background. If you cannot choose white, a lighter yellow or blue, or a similar color, can be acceptable.



- Light text on a dark background can be eye-catching, but should be used sparingly because it can fatigue the eye. Keep the text large enough to be readable.
- Avoid patterned text, such as stripes or complex designs. This will make the text more difficult to read. Also avoid putting type on patterned backgrounds for the same reason.
- Keep in mind who your target audience is. If you are designing a website simply for the graphic design community, you can take more chances than you could with a site designed for the general public.

# Creating Accessible Web Design and Typography

---

Every designer and developer must consider accessibility when creating for the Web. Following the rules of accessible (or Section 508–compliant) web design is key to a successful website for all users, and is imperative for all governmental websites.

Section 508 compliance stems from the Workforce Rehabilitation Act of 1973. It requires that electronic and information technology that is developed or purchased by federal agencies be accessible by people with disabilities.

Vision-impaired individuals use a machine called a “screen reader” to convert the text of a website into spoken words. The screen reader uses software that ignores colors and fonts, and simply reads the text to the user.

You should keep this in mind when designing for the Web. Every image should have an “alt tag,” a word or short phrase that will be used by the screen reader to describe what the image was intended to convey.

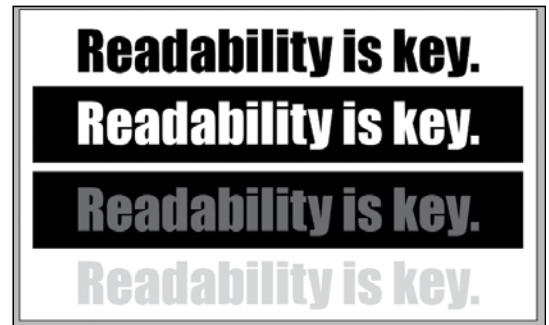


## Some Tips for Designing for Accessibility

Here are a few suggestions for designing with accessibility in mind:

- For best results, be sure to use CSS to specify font size, family, and other font variations. This includes underlining, italics, bolding, and capitalization. It is very helpful for screen readers if all text is styled with a CSS style.

- Don't use color as the only way of conveying information; make sure that the message of your web pages would still be clear if all colors were removed.
- For visually impaired people who will be viewing your pages, but do not necessarily need to use a screen reader, try to avoid text and backgrounds that are too similar in color. For example, black text on a very dark gray background might be difficult for some people to read. In the example shown here, you will notice that some versions of this text are more readable than others.



## Conclusion

Typography on the Web has truly come a long way, but there is still a long road to travel. Take the time to do your research and stay informed on the latest trends. Joining user groups, searching online for updated information, and talking to folks who are in the trenches of web design are all good ways to keep abreast of the current trends in the world of web typography.

Remember to test your content on multiple browsers and on different platforms (both Mac and Windows are highly recommended). Not all browsers display type the same way, and user-defined settings on an individual user's computer can give your web pages a completely different look than you intended. Take time to inspect your pages and educate yourself so you can reach your full potential and be the best web designer/developer you can be.





# Glossary

# Glossary

**ACCESSIBILITY** Refers to the degree to which content on a page can be accessed and used by an audience. The idea is to make information accessible to individuals who would otherwise not be able to use the Web.

**ALIGNMENT** The positioning of page elements at the left, right, center, top, or bottom relative to each other.

**ALT TEXT** A short, text-only description of an image that describes the image for users with disabilities who use screen readers or users who have images turned off in their browsers.

**ASCII** Abbreviation for “American Standard Code for Information Interchange,” a standard originally developed for data transmission via Telex. Each character is represented by a numeric code, allowing data exchange across different systems.

**BANDWIDTH** In the digital world, this refers to the data rate (amount per unit of time) that can be processed in real time by a given device or program. As an example, a dial-up modem has a very narrow bandwidth compared to a cable modem or a LAN connection, so the former is not suitable for streaming high-bandwidth videos on the Internet. Originally, the term comes from analog signal transmission, describing the width of the electromagnetic frequency range or “band” a signal needs for its transmission.

**BREADCRUMB NAVIGATION** A form of website navigation that enables the user to work out the depth to which they have navigated within a website or a section of a website.

**BROWSER** An application used to retrieve data from the Internet and to process and display it on a client computer. In addition to text and HTML, modern

browsers also display graphics, movie clips, and various other types of media, frequently using plug-ins for these tasks.

**BYTE** A data unit equal to 8 bits.

**CSS (CASCADING STYLE SHEETS)** A set of rules to control the appearance of your document. When you use HTML with style sheets, the HTML document contains structured content, whereas the information on the document’s appearance resides in a style sheet. By extracting the design information, the HTML code is considerably simplified. Design elements in a style sheet can also easily be applied to other HTML files.

**CSS CLASS** A syntax for specifying a CSS selector by a class attribute of an element. The name of a class selector in style rules is preceded by a period (.). Class styles can be used more than once on a page and are helpful in bringing consistency to your web design.

**CSS ID** A syntax for specifying a CSS selector by a unique attribute of an element. The name of an ID selector in a style rule is preceded by a pound symbol (#).

**CSS PROPERTY** A named style attribute or parameter for a markup element specified in a style sheet declaration, such as color or font-family, which is assigned a specific value.

**CSS SELECTOR** An identifier that determines which style declarations are associated with specific elements of a document. Multiple selectors can be grouped together in the same rule by separating them with commas.

**CSS VALUE** A quality assigned to a style property. CSS properties each have specific value ranges that can be declared for them. A value is associated with a property; together, they form a declaration.

**DATABASE** A way to store data in tables so that it can be accessed in a very efficient manner. There are many different types of databases, including MySQL, SQL Server, Oracle, and Access.

**DEPENDENT FILES** Any files used on your website other than text content. Dependent files can be style sheets, movie clips, bitmap images, JavaScript files, sound files, and so on.

**DIV TAG** A tag used to define a division or a section in an HTML page. It is frequently used to group elements so they can be formatted with the same styles.

**FILE TRANSFER PROTOCOL (FTP)** A method of uploading and downloading files to and from an Internet server.

**FLOWCHART** A graphical representation of how information flows within a computer, a program, or an interactive process on a website.

**FONT** The complete character set (numbers, uppercase and lowercase letters, and in some cases, small caps and symbols) of a particular typeface in a specific style, such as Verdana Bold.

**FORM VALIDATION** A process that is employed to validate user input in HTML forms before sending the data off to a server. It can be performed either client-side (using JavaScript or Spry) or server-side (using a server-side language such as PHP).

**FORM VARIABLE** A particular type of variable that is created when a user fills out a form that uses the POST method. Once the user clicks the Submit button in a form, each value they have chosen on the form becomes a form variable. The name of the form variable is the name of the form field, and the value of the variable is whatever value has been assigned to that form field or written into the form field by the user.

**GET** An FTP command telling the client to download a file from the server to a local drive. To make sure you are editing the current live content of a page, use GET each time you want to edit a page from the server.

**GRAPHICAL USER INTERFACE (GUI)** A software component that allows the user to interact with the computer by means of graphical elements. The displayed screen elements can be accessed with a pointing device (in most cases, a mouse).

**HEX VALUES** Numbers in the hexadecimal system, which are commonly used for specifying colors on web pages.

**HTML (HYPERTEXT MARKUP LANGUAGE)** A tag-based language that allows content to be delivered over the World Wide Web and viewed by a browser.

**HYPERLINK** The clickable part of a graphic or web page text that directs the user to another web page, a download, or another website.

**IMAGE MAP** An image that has several hyperlinked areas (or “hotspots”) on it.

**JAVASCRIPT** A scripting language that adds special functionality and features to dynamic HTML pages. JavaScript code is nested within an HTML page and, unlike CGI scripts, is executed inside the browser.

**META TAG** A tag in the header of an HTML page that contains information not directly addressed to the user. Meta tags are read and processed by browsers and search engines; however, today’s search engines focus more on page content, ranked pages, and linked pages than on meta tags.

**PHP** An open source, server-side scripting language used to create dynamic web pages. See also “server-side language (SSL).”

# Glossary

**PLUG-INS** Browser modules that are used to recognize and display certain file formats. For example, plug-ins are used to help display the 3D data of audio files.

**REMOTE SERVER** A network computer that stores your website files and makes them accessible to the public via a local or wide area network, usually applying the HTTP and TCP/IP protocols.

**SERVER-SIDE FORM VALIDATION** A way to validate a form after the form has been sent to the web server by the user. A server-side language such as PHP, ColdFusion, or ASP is used to create a script, which checks the data that the user put into the form to be sure that the data is in the correct format. See also “form validation.”

**SERVER-SIDE LANGUAGE (SSL)** A language that executes script code directly on the server to fulfill a user request. Interactive websites relying on databases often employ an SSL. In contrast, client-side scripting languages like JavaScript are run inside the user’s browser.

**SITE MAP** A representation of a website’s structure, usually an HTML page that displays and links to all of the site’s content. It can be used as a planning tool for web design or as a guide and overview for the visitor to a website. Search engine bots can also take advantage of an online site map.

**SQL (STRUCTURED QUERY LANGUAGE)** The language of the database. SQL includes four main statements that are used to pull data from a database, insert data into the database, update data stored in the database, or delete information from the database.

**STORYBOARD** A series of sketches depicting a process, such as the steps in a user’s interaction with a computer program or a website. Storyboards are useful during the planning phase of a project, for presentations, and to check the steps of a process for logical errors.

**TAG** A formatting command that designates the start or end of an element. Tags specify how an HTML document should be formatted.

**TRANSPARENCY** The property of an image element with an opacity value below 100 percent that allows parts of background elements or entire background elements to show through.

**URL (UNIFORM RESOURCE LOCATOR)** The address of an Internet resource. It contains the transmission protocol, such as “http” for web pages or “ftp” for file transfer; the server address, including the domain; an optional port; and a path on the remote server.

**URL VARIABLE** A variable that is created using a querystring or by submitting a form using the GET method. In any server-side language you need to reference a URL variable in a specific way.

**VALIDATION** The process of checking that code written in a web language like HTML or CSS complies with the standard for that language.

**VARIABLE** A way to store a value in computer memory under a single name for a particular length of time. There are different types of variables that last for different lengths of time and can be created in different ways. The value of a variable can change over time, but the variable name does not change.

**WEB BROWSER** See “browser.”

**WEB-SAFE COLORS** In the past, many computer systems could only display up to 256 colors. To cope with this, a fixed palette of colors was introduced that most systems were capable of displaying. Web-safe colors are not really relevant today, since even somewhat dated computer systems can now display thousands or millions of colors.

**WIREFRAME** A stripped-down, no-content version of a website that is a diagrammatic representation of the site’s page structure and navigation. A wireframe consists of cross-linked pages that behave like the final website without the graphics or final text content.

**XML (EXTENSIBLE MARKUP LANGUAGE)** A standard description language. XML files consist of pure ASCII data and they can be edited using any text editor. The benefit of XML compared to other description languages like HTML is the strict distinction between layout and structure.





# LEARN BY VIDEO

*Revolutionize the way you learn™*



Android App Development and Design:  
Learn by Video  
(ISBN 9780321786807)



Image Editing with Camera Raw in Adobe  
Photoshop CS5: Learn by Video  
(ISBN 9780321786883)



Make Your First Video with iMovie '11:  
Learn by Video  
(ISBN 9780321786906)

Are you looking to get up to speed on the latest creative software, master essential techniques in photography and design, or just build your knowledge of today's most important technology topics? Then don't miss the new innovative **Learn by Video** training workshops from video2brain and Peachpit Press.

**Learn by Video** workshops offer up to 5 hours of focused, exceptional-quality video tutorials presented by experienced instructors, as well as lesson files, assessment quizzes, and review materials. The DVD is bundled with a printed booklet that provides supplemental material to enhance the video training.

Crisp, 1280x720 High-Def video that you can watch in full-screen mode for an exceptionally immersive learning experience

High-quality downloadable exercise files are included on the DVD

Bookmark your favorite videos

Table of Contents never more than a click away

Watch-and-Work mode shrinks the video into a small window so you can practice while you learn

Video player remembers which movie you watched last

Full-color 48-page booklet inside

## Additional Titles

- iPhone and iOS 4 Advanced App Development: Learn by Video (ISBN: 9780321786777)
- Work Like a Pro Photographer in Aperture 3: Learn by Video (ISBN: 9780321786814)
- Using Layer Masks for Image Editing in Adobe Photoshop CS5: Learn by Video (ISBN: 9780321786890)
- Use Facebook to Promote Your Business: Learn by Video (ISBN: 9780321786784)
- Mastering Exposure in Digital Photography: Learn by Video (ISBN: 9780321786937)
- Building a Data-Driven Website with Adobe Dreamweaver CS5.5: Learn By Video (ISBN: 9780321786838)
- What's New in Adobe Flash Professional CS5.5 and ActionScript 3.0 Learn By Video (ISBN: 9780321786821)
- Adobe InDesign for Creating eBooks: Learn by Video (ISBN: 9780321786845)
- Introducing Adobe Audition CS5.5: Learn by Video (ISBN: 9780321786852)

For a full list of Learn by Video titles, go to [www.learnbyvideo.com](http://www.learnbyvideo.com)





# WATCH READ CREATE

Unlimited online access to all Peachpit, Adobe Press, Apple Training and New Riders videos and books, as well as content from other leading publishers including: O'Reilly Media, Focal Press, Sams, Que, Total Training, John Wiley & Sons, Course Technology PTR, Class on Demand, VTC and more.

No time commitment or contract required! Sign up for one month or a year. All for \$19.99 a month

**SIGN UP TODAY**  
[peachpit.com/creativeedge](http://peachpit.com/creativeedge)

creative  
edge