

Greg's Guide to Developing Online Learning Courses

By [Greg Kearsley](#)



Introduction

This Guide covers the essential things you need to know to develop online (web-based) learning courses. It presumes a little familiarity with the internet/web, but not much. It also assumes some prior experience with online learning courses.

Be forewarned that developing courses for the web can get pretty complicated because there are, ultimately, many arcane details you have to understand. On the other hand, you can get a lot accomplished with just a little bit of knowledge. This guide aims to provide just enough information to make you dangerous.

For more detailed guides/references to the topics mentioned here, see the Resources section at the end of this document.



The Basics: HTML

While it is quite possible to develop complete online courses without any knowledge of HTML (see following sections on Web Editors and Integrated Learning Systems), it is difficult to do any advanced stuff without some understanding of HTML.

HTML is simply a formatting language with commands for specifying the characteristics of text (e.g., size, fonts, color), creating tables, or page/screen layout. However, it has one very important capability: links. In HTML you can specify links to any document (i.e., file) on any computer (i.e., server) that is part of the internet. Those files are usually text documents, but they can be multimedia (i.e., audio/video) in nature too – which makes it interesting. To place a link to a document, you use the following HTML command:

```
<A href="http://www.myserver.com/mydocument.htm">Title of the link</a>
```

For online introductions to HTML, see:

http://hotwired.lycos.com/webmonkey/authoring/html_basics

<http://www.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimer.html>

<http://www.davesite.com/webstation/html>

<http://www.weballey.net>

P.S. If you'd like to start peeking at the HTML used to create the web documents you are viewing, just select the "Source" option from your Browser View menu.



Beyond the Basics: Javascript and friends

HTML alone won't provide any interactivity (other than the capability to link to other sites). Javascript is a web scripting language that allows you to process input and do something with it. The most common use of Javascript is to create forms with input fields. But you can use Javascript to write any kind of interactive sequence, such as quizzes or calculations. The catch is that you cannot use Javascript to save data – for that you must pass data to a program residing on a server written in Perl, C, or a general programming/database language. BTW, Javascript is not the same as Java, the latter being a distinctly different web programming language used to write “applets” – programs that can be loaded by a web document. (confusing, eh?) You might also run across something called DHTML (Dynamic HTML) – this is an extended form of HTML that allows you to create movement (animations) on a web page.

To learn more about Javascript, see:

<http://www.zdnet.com/devhead/filters/homepage>

<http://www.javascriptguide.com>

<http://home.netscape.com/eng/mozilla/3.0/handbook/javascript/index.html>

<http://javascript.internet.com>



Web Editors: The easy way to create web documents

Luckily you don't actually have to learn HTML or Javascript at all to create web documents. This miracle is accomplished through the use of a web editor – which works like a word processor, except it saves everything as web files. There are many web editors available; some popular examples include Dreamweaver from Macromedia, Microsoft FrontPage, and GoLive CyberStudio. Actually, current versions of word processors, such as Microsoft Word, let you save your documents in HTML form (use save as web document), so you don't need to have a web editor for simple jobs. But if you are going to do serious web development, you are going to want to have/use a web editor.

Even though using a web editor relieves you of having to create HTML manually, it still helps to understand a little about HTML in order to figure out and troubleshoot what web editors do. This is especially true once you start doing more advanced web documents that include multimedia or interactivity (via Javascript).

For more about web editors, see:

<http://builder.cnet.com/Authoring/WysiComparo>

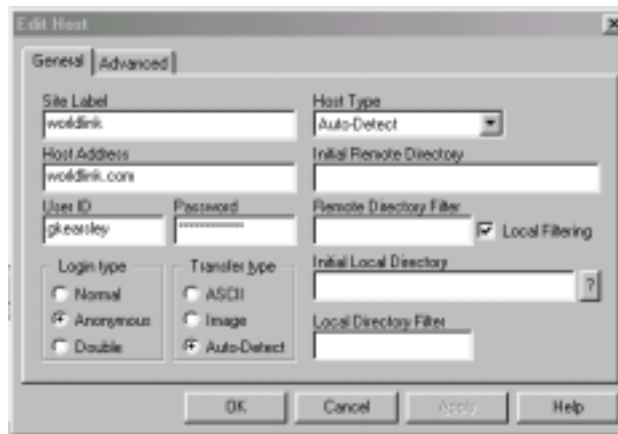
<http://www.webcommando.com/editrev/index.html>

<http://www.zdnet.com/devhead/stories/articles/0,4413,2193639,00.html>

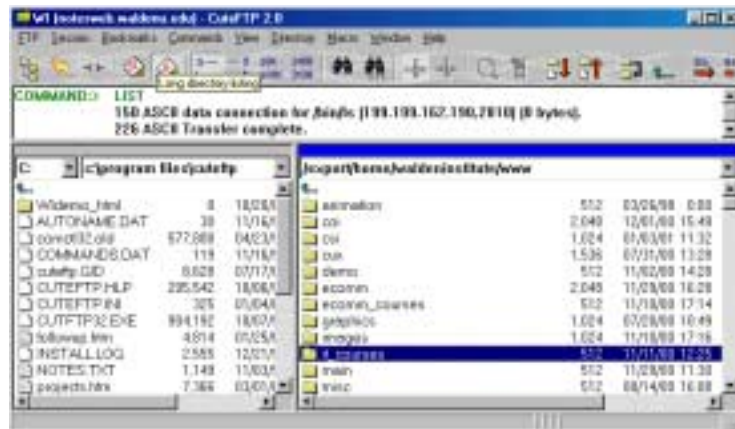
FTP: The basic workhorse of the web

Since all web documents are files that reside on a server somewhere, and you create these files on your PC, you need a way to upload/download files between the server and your PC. This is accomplished using an FTP (File Transfer Protocol) program. Unfortunately, many Windows systems do not come with an FTP program as part of their standard utilities, so you have to procure one such as WS-FTP (<http://www.ipswitch.com>) or CuteFTP (<http://www.cuteftp.com>) -- both of which have free trial versions. Many ISPs provide a customized FTP capability (usually called “publish to the web”) as part of their software package.

Once you have an FTP program, you need to set it up for each server you plan to transfer files to/from. This involves entering the server address, plus your user id/password for the server. If you don't have an account on the server, you're out of luck – unless it happens to accept “anonymous FTP” requests, which means that no id/password is needed. Many universities and other organizations run anonymous FTP sites for public sharing of files. However, for your own web documents, you probably want to store them on a private site, protected by an id/password. Most ISPs (Internet Service Providers) provide server space with your account, and you should be able to get an account on your organization's server by going through the IT dept.



Once you have your server entry set up, transferring files back and forth is just a matter of selecting the directories where files are located or to be moved to, and clicking on one or more files to transfer them. (The window on one side shows the files on your PC and the window on the other side lists the files on the server.)



Note that current versions of browsers let you enter an FTP command directly in the location field as a URL. In other words, you can type in: “ftp://www.myserver.com” and it will prompt you for your id/password. You can then use the “Upload file” command from the File menu (Netscape) and upload your file to the server. While this will do in an emergency, you really need a FTP program for serious file transfer work.



Graphics & Sound Editors: Simple Multimedia

What happens when you want to create or work with pictures, illustrations, or doodles? You need to have a graphics editing program – of which the industry standards are Adobe Photoshop and Illustrator (two separate products). CorelDraw and Macromedia Freehand are other popular graphics editors. These programs allow you to manipulate photos (e.g., resize, crop, rotate) or draw illustrations from scratch. They also accept input from scanners or digital cameras. Many images are available in clip art collections on the web or CD-ROM. BTW, this is a good place to remind everyone that while it is simple to grab any image you see in a web page with a right click and “Save Image As” command, it is not legal to do this. If you take an image from another site, you must request permission to use it. That’s why clip art collections are valuable – once you have bought their image collections, you are free to use them as much as you want.

Another important use of graphics editors is to change graphics formats. There are a number of different file formats for images and only two work universally for web documents – namely “.gif” and “jpeg”. So if you get an image in another format (e.g., “.bmp” or “.tiff”), you can normally open it with a graphics editor and resave it in web-friendly formats.

While we’re on the topic of graphics, I should mention a very useful tool to have – screen capture programs. These programs let you capture any portion of the screen you are looking at and save it as a graphics image – which you can then manipulate further using a graphics editor. This is how you get screen images to include in a document or web page (such as those in the previous section). The one I use is Snag-It (<http://www.snagit.com>) although there are many of them.

There is quite a lot to know about the design and preparation of images for the web (don’t even mention color), and if you intend to do a lot (or even a little) work with graphics, you should check out one of the books by Lynda Weinman (e.g., <http://www.lynda.com/dwg3>).

Just as there are graphics editors for images, there are sound editors to manipulate audio files (voice, music, sound effects). These editors let you chop up, change volumes or add special effects to audio clips – as well as let you record directly to your machine. The

most commonly used product is Sound Forge from Sonic Foundry. Sound editors are also useful in converting audio file formats – although most formats will work on the web these days.

Once you have create a graphics or sound file, it is easy to get them to work on the web. After you have FTP'd them to your server, you simply call them with an tag for graphics or <A HREF> tag for sound. This assumes you know your HTML -- or are using a web editor (in which case you just insert them where you want them).



Animation and Video: Fancy Multimedia

Creating animations and video sequences for web documents involves the same process as graphics or audio – you just need suitable editing tools. The industry standard tool for creating web animations is Flash from Macromedia. There is also a program called Director from Macromedia that lets you create stand-alone animations (i.e., for CD-ROM delivery) and then convert them to web versions using a program called Shockwave. These are sophisticated programs that take a while to master; however, they don't involve any programming. I am not a big fan of animations in learning programs since they take control away from the user – unless they are demonstrating a dynamic process or procedure.

Earlier it was mentioned that you can create relatively simple animations in DHTML using a web editor. You can also create simple animations with sound using Microsoft Powerpoint and distribute the slideshows using the “show” (.pps) format which doesn't require that the user have the Powerpoint program. However, such shows will have to be downloaded by the user; they can't play in real-time.

Doing video for the web is a little more complicated since you first need to obtain the video footage. There are a variety of ways to do this: (1) you can use a traditional (analog tape) video camera and then digitize the tape from a VCR using a sound capture system such as Dazzle (<http://www.dazzle.com>). [Note that most Macintoshes have this capability built-in.] (2) you can capture the footage in digital form (mpeg format) using a digital video camera and load it directly into your machine, (3) you can use a inexpensive “videocam” that mounts on top of your PC to produce digital video of anything you can get in front of your machine.

Once you have the video captured, you need to use a video editing program to manipulate the clips (unless you want to use them exactly as they were captured). There are many such programs of which Adobe Premiere and Ulead MediaStudio are the most commonly used. Normally you want to put video to be used on the web in “streaming” (compressed) format which them requires a browser plug-in to play. The most common streaming

formats are RealMedia, Quicktime, and AVI (Windows Media). To learn more about streaming video, and see some examples, visit <http://pocahontus.doit.wisc.edu/index.html>

Besides the standard animation and video editing programs, there are some other interesting possibilities. RealMedia (<http://www.real.com>) produces a number of products that let you put slideshows with audio in streaming form for web delivery. There is also a program called Camtasia (<http://www.camtasia.com>) that lets you capture a sequence of screens from any application running on your system and produces a “screen movie”. This is useful for creating a tour or demonstration of a program or web site.

One thing to keep in mind when creating any multimedia material is that they produce large files (especially video) which makes for lengthy uploading/downloading times and requires a lot of server space. If you are going to do a lot of multimedia in a course (or even just have a lot of large text files), it’s a good idea to consider using CD-ROM to distribute material offline. CD-Writable drives and blank CDs are quite inexpensive these days – and you can always have them duplicated at Kinkos.

A warning: doing animations and video on the web requires a lot of knowledge, skill and practice. It is definitely an advanced design topic!



Integrated Learning Systems

Most course developers and faculty creating an online course today are likely to use an integrated learning system such as Blackboard (<http://www.blackboard.com>), WebCT (<http://www.webct.com>) or LearningSpace (<http://www.learningspace.com>). You can find comparisons of these and other systems at: <http://www.ctt.bc.ca/landonline/choices.html>, <http://multimedia.marshall.edu/cit/webct/compare/index.htm> and <http://www.outreach.utk.edu/weblearning>

Integrated learning systems provide all the tools needed to develop and deliver online courses including content formatting, quiz editors, student grading/progress tracking, as well as built-in discussion form, email, chat, and file management capabilities. Furthermore, they have easy to use interfaces that eliminate the need to do any programming (or know HTML).

So why do we need anything else? For one thing, these systems tend to be all or none, i.e., you must buy everything included in the package, even if you only need/want some of the capabilities. Secondly, the ease of use achieved by the built-in formatting and layouts comes at the expense of flexibility – you can only design screens and controls within the options provided – which tends to be pretty limited. Third, the capabilities provided by all the sub-components (e.g., quiz editors, email, chats) tend to be less

sophisticated than stand-alone programs available. Finally, you still need to use many of the tools discussed in this guide to produce multimedia elements.

Integrated learning systems are an excellent choice for novice developers or faculty who want to create online courses quickly with a minimum of knowledge required. On the other hand, most experienced developers will prefer to use stand-alone tools for the flexibility and power they afford.



Design Considerations

So far, all of the discussion in this guide has focused on the tools available to create web content – which is certainly the main pre-occupation of a course developer. Beyond learning how to use the tools is actual design knowledge and skill. For online courses, this can be divided into three categories: instruction, usability, and aesthetics. Instructional design involves tasks such as the assessment of learner needs and background, definition of learning objectives, identification of appropriate learning/teaching strategies, creation of suitable learning assessment methods, and media choices. Most aspects of instructional design for an online course need to take into account the particulars of the delivery system to be used since this determines the nature of the learning/teaching environment,

Usability concerns how easy it is for the student and teacher to learn or teach online. Most web-based course delivery systems (especially the integrated learning systems) have very good user interfaces and high usability. Furthermore, since these systems do not involve any user programming, they are relatively free of bugs and hence highly reliable. On the other hand, if you create your own interface in HTML, or use multimedia elements in your courses, you will have to pay a lot of attention to interface design and usability. Broken links are one of the continuing aggravations of web-based courses and require ongoing monitoring by the course developer or faculty member.

Aesthetic aspects of design have to do with screen layout, choice of fonts, color, use of illustrations/photos, and presentation form (e.g., metaphors). Aesthetic factors mostly contribute to the appeal or attractiveness of a web site, but some factors also have usability or learning implications – such as font or color choices which affect legibility and comprehension. There are a number of style guides that provide do's and don'ts for web design. (For fun, see Vincent Flander's <http://www.webpagethatsuck.com> site.)

In the design of any online course, you have to take into account all three categories of design – even though they sometimes conflict. For example, from an aesthetic perspective you may want to use very little text on a screen or some decorative illustrations, but from a usability point of view, this may distract or slow down the user.

Design is all about making trade-offs, and you continually have to balance these three design categories.

Finally, the principle of “universal design” should be mentioned – creating online courses and web materials that are usable by all individuals, including those with disabilities. This involves making sure that any multimedia elements have alternate captions, i.e., images, audio and video have text descriptions, as well as providing text-based screens for pages that use frames or have complicated graphic layouts. More information on universal design can be found at these sites: <http://www.cast.org>, <http://www.webable.com> and <http://www.rit.edu/~easi> .



Resources (a short list)

Horton, W. Designing Web-Based Training. Wiley, 2000.

- A good introduction to the design and delivery of online courses focusing on the training realm.



Powell, T. Web Design: The Complete Reference. Osborne/McGraw-Hill, 2000.

- Comprehensive discussion of the issues involved in web design

Niederst, J. Web Design in a Nutshell. O'Reilly, 1998.

- Reference guide to use of HTML and audio/video files for the web



Nielsen, J. Designing Web Usability: The Practice of Simplicity. New Riders, 1999.

- Excellent discussion of all the issues associated with interface design for the web. Nielsen's web site is <http://www.useit.com>

Lee, W. & Owens, D. Multimedia-Based Instructional Design. Jossey-Bass, 2000.

- Good instruction to instructional design for multimedia



Lynch, P. & Horton, S. Web Style Guide. Yale Univ Press, 1999.

- Guidelines to follow in the design of web documents. The web site that corresponds to this book is <http://www.info.med.yale.edu/caim/manual>