

How To Guides for the WorldWide Web

"How To" Guides for the WorldWide Web

[Web Browsers](#)

How to guides for web browsers such as [Firefox](#).

This is an automatically-generated printer-friendly PDF version of the HTML web page at [How To Guides for the WorldWide Web](#) ← CLICK HERE to get the standard HTML version of this document.

This is an automatically-generated printer-friendly PDF version of the HTML web pages at [How To Guides for the WorldWide Web](#) ← CLICK HERE to get the standard HTML version of this document.

Web Search Tricks

Tricks to Use with Web Search Portals

Tired of the same old sites at the top of the results of many web searches? Here's a trick to use to get some different results.

Start with just a few words describing what you are search for. Remove all of the spaces, punctuation and capitalization, and then search for those words, all run together. The reason that this works so well is many users are tagging sites with tags that consist of simply a string of characters.

For example, if you are searching for Penn & Teller: Bullshit! remove the spaces, punctuation and capitalization, and simply search for penntellerbullshit.

Try it!

This is an automatically-generated printer-friendly PDF version of the HTML web pages at [How To Search the Web](#) ← CLICK HERE to get the standard HTML version of this document.

Choosing a web site design firm

Here are some considerations for choosing a company for designing a web site:

What level of [HTML](#) will be used?

The site designer you choose should be using some of the latest [web publishing standards](#). For some reason, however, quite a few web developers are still stuck designing with the 1997 [HTML](#) 4.x standard. In Internet time, this is ancient, so you'll want to watch out for this.

To determine which version of [HTML](#) that a designer uses:

Using your web browser, go to the home page of a web site they have recently developed.

Select the option to view the page source.

Look at the <html ...> tag.

If the <HTML> tag, or any other tags are in upper case, or it has no attributes, then the site was developed with the 1997 [HTML](#) 4.x standard. That might be good enough if the users you are targeting were still using Internet Explorer version 4, but this is highly unlikely.

If the page was developed for more recent [HTML](#) standards, the <html ...> tag should be in lower case and have some xmlns attributes. One of the attributes should be a default namespace declaration (an xmlns attribute that does not declare a prefix), which looks like one of these:

```
xmlns="http://www.w3.org/1999/xhtml"
```

```
xmlns="http://www.w3.org/2002/06/xhtml12"
```

The 4-digit year in the namespace declaration does not indicate the level of [HTML](#) standard, just the year that the namespace for the initial release of that version of [HTML](#) was assigned.

How good is their coding?

To see how error-free the coding is, submit the URL of the site's home page to one of the online validators:

[HTML](#) or [XHTML](#) only

If it's an [HTML](#) or [XHTML](#) site, submit the designer's home page URL to the [W3C Markup Validation Service](#).

XML

If it's a site that uses XML web pages, such as this one, (... we need a link to an XML validation service).

How To Guides for the WorldWide Web

Mobile

If you want your web site to support cell phones and other mobile devices, such as smart phones and PDAs, use [MobiReady Report](#) to analyze the designer's home page or your own site. For an example of the report, submit sample.mobi to that form.

Do they understand how to build interactive web sites?

Too many web design professionals can build good static web sites, but don't understand the intricacies of building an interactive web site using dynamic pages generated by application servers. Too many times, web sites do not work properly because the developers did not fully understand how to design for a client-server environment, as is required for the interaction between a web browser and an application server. In particular, close attention must be paid to which pieces of data are stored on the client side of the connection and which are stored in the session on the application server. For example:

If the prices in a shopping cart are stored on the client side of the connection, then a knowledgeable user could potentially alter the prices of items in the cart before they were submitted to the server, essentially giving themselves a nice big discount.

One very popular dating web site stored the ID of the profile that you most recently viewed on the server, rather than on the client where their web-based e-mail form was being displayed in the browser. So if you had been viewing more than one person's profile, there was a good chance that the message you were entering for one person would actually be sent to another person!

How to tell if a developer understands client-server programming

To determine if a web site performs as expected under normal user interaction would take extensive testing and depends on the functions of the web site itself. However, there is a straightforward test, using the login and logout functions, on most interactive sites, that can be used to determine if those functions will work properly under more normal circumstances:

Ask the developer for an example of an interactive web site that they have developed and you can try out yourself.

Register for an account on the site to get a username and password.

Log into the site using your username and password. This should start a session that persists until you either logout or close your browser.

Right-click on a link and open another page on the site in a new window. You now have two windows open that should be using the same session credentials.

Go back to the first window and logout, or else wait however long it takes for your session to be timed out.

Go to the second window again and navigate to a page where some personal, user-specific information would be displayed. The site should display a message indicating that your session has timed out and that you need to log in again. If you still see user-specific information at this point, then the server side of the session did not get reset.

Close the second window and go back to the first one.

Right click on a link, the one to "login" if possible, and open it in a new window. Go to the login page and log into the site again in the second window.

Close the second window and go back to the first one.

Right click to open the second window again. Navigate to a page where some user-specific information would appear and make sure you are still logged in. If it seems that you got logged out, then the client-side session credentials were not properly stored in the browser session.

Web Publishing Standards

Web publishing standards

Web publishing standards can be separated into two categories:

Those standards that apply to the client browser, and must be supported at least to some extent in any browser
Standards that apply to the server, and therefore may or may not be applicable depending on whether the server delivers static web pages or dynamically generated content (via CGI, servlets, portlets, etc.), and will vary in level of support provided by different products depending on whether the vendor provides proprietary or open-source solutions

Advantages of designing web sites with the latest standards

Faster page loading

Putting the common content of your site's pages into separate [template files](#), allows a browser to download the templates into cache just once for the entire site. Since the common elements can be half of the [HTML](#) code for typical web pages, moving them into templates can easily reduce the size of the files being downloaded by 50% or more.

How To Guides for the WorldWide Web

Beware of some older design tools and web developers that duplicate the common elements on every single web page. This slows down page load times, increases bandwidth requirements, and makes the site much harder to maintain.

Since your site is probably being developed and tested over a high-speed LAN or broadband connection, it's likely that you would not even notice a performance problem that could seriously affect users who are not on a similar type of connection.

Retain more visitors

The longer pages take to load, the more likely that web site visitors will leave and go to another site. When your pages load faster, fewer visitors will be leaving your site to go to another, faster one.

Some might argue that since most users now have broadband connections, you can probably assume that users have high-speed Internet connections, and so page download time doesn't matter much any more. But this misses the fact that the number of cell phones and other wireless devices accessing the Internet with slower connection speeds has been continuously increasing. So you shouldn't assume that users have a fast connection because connection speeds are actually all over the place. And do you really want to lose visitors on slower connections, even if you assumed those users were only the ones on dial-up connections?

Lower bandwidth costs

When the common elements of a web site's content can be cached, the bandwidth requirements and the associated costs will be greatly reduced. If instead those elements are put into web pages on the server side and downloaded with every web page, page sizes and bandwidth requirements can easily be doubled.

Easier to maintain

When the common elements of your content are stored in one place, it makes your web site much easier to maintain. To change the look and feel of your site, you simply change the template in the one file containing the common elements. Changing individual pages is also easier, since those pages are not cluttered up with all of the common site elements.

Cross-browser compatibility

Sites designed for more recent standards are more likely to look pretty much the same in all current browsers. In addition, if the pages are small enough, they may even look acceptable when displayed on portable handheld devices.

Forward compatibility

As standards evolve, new versions are developed with backward compatibility in mind. When necessary for a change that introduces compatibility problems, those changes are typically made gradually over a number of versions and by gracefully deprecating the older specifications. Developing web sites to more recent standards will reduce the work needed to update a site as the standards evolve.

This discussion applies to [web publishing standards for browsers](#), which are on the client side of an [HTTP](#) connection. From the browser's point of view, it doesn't matter how the pages are constructed on the server, whether they are static pages that are simply downloaded or dynamic web pages that are built on-the-fly by an application server. So you can easily tell whether a site is using templates by viewing the source after downloading a page in a browser. If, above the html tag near the top of the page, you see one or more xml-stylesheet processing instructions such as any of these examples:

```
<?xml-stylesheet type="text/xsl" href="/templates.xsl"?>
```

```
<?xml-stylesheet type="application/xslt+xml" href="/templates.xsl"?>
```

then the site is using templates. If, on the other hand, there is no xml-stylesheet instruction before the html tag, you'll probably see [HTML](#) for common elements in the source for that page as well as duplicated in other pages all over the site, which indicates that there is a lot of wasted code and bandwidth.

Since there's no good reason at this point to be designing web sites that need to support 1997 [HTML](#) 4.x browsers, such as Internet Explorer version 4, using more recent standards provides all advantages and no disadvantages. So this should be a no-brainer unless you really want your site to cost twice as much for bandwidth, be more difficult to maintain and to lose visitors.

Web publishing techniques

Publishing techniques for web sites have evolved almost as much as the [web publishing standards](#) have evolved. Since the protocols that allow clients and servers to communicate on the worldwide web are well defined and designed for interoperability, things can be simplified by looking at just one side or the other at a time. For example,

How To Guides for the WorldWide Web

when looking at web publishing from the perspective of content being delivered to the client browser, the wide variety of ways that a server can produce content can be ignored. From the client perspective, the evolution of web publishing techniques from the simplest to the most sophisticated would include:

Choose a web design tool that supports separate templates

When selecting a web design tool, make sure that it supports creating site templates that are stored in separate files, which can be downloaded and cached separately by client browsers. Many web design tools apply the templates to every single web page before the pages are published. An application server might be apply site templates to every page it sends, which takes more time on the server and can more than double the size of the pages being sent to the client. If you are already using a web design tool or application server that works this way instead of sending its templates to the client as separate files, you will probably want to [create separate templates](#) yourself, unless your web server bandwidth is free and you don't mind the longer page load times. The only disadvantage is that by putting the common elements of your web pages in a separate file, the web pages might not be quite WYSIWYG in the design tool; you will need to open the files in a web browser to see the pages as they would look on the actual site.

This is an automatically-generated printer-friendly PDF version of the HTML web pages at [Web Site Designers](#) ← CLICK HERE to get the standard HTML version of this document.

[W3C Markup Validation Service](#).

[XML](#)

[MobiReady Report](#)

This is an automatically-generated printer-friendly PDF version of the HTML web pages at [HTML and XHTML](#) ← CLICK HERE to get the standard HTML version of this document.

How to change your web site's favorites icon

The easiest way to get a favorites icon is to download one, from a web site such as [Artscape](#) and then rename it to favicon.ico.

The alternative is to create a 32x32 256-color image in a picture editor, and convert it to an icon file. In some image editors, the conversion can be done using a plug-in; otherwise there are some free programs that can do it.

Upload the favicon.ico file in binary format to the root directory (public_html) of your web site.

Update the favorites icon in the browser

To load the new favorites icon in FireFox:

Browse to the URL of the icon itself: <http://www.yourdomain.com/favicon.ico>

To load the new favorites icon in Internet Explorer:

On the menu bar, select Tools then Delete Browsing History....

Select Delete files... to delete Temporary Internet files.

Confirm by clicking Yes.

Close down all instances of the browser.

Start the browser again and browse to the site's home page.

If the favorites icon changes to an Internet Explorer icon, shut down the browser again, restart it one more time, and browse to the site's home page.

This is an automatically-generated printer-friendly PDF version of the HTML web pages at [Favorites Icon](#) ← CLICK HERE to get the standard HTML version of this document.

How to integrate a web site with various bookmarking and tagging sites

Bookmark This

Bookmark a Site

`Bookmark This Site`

Example: [Bookmark This Site](#)

Bookmark a Specific Web Page

`Bookmark This Page`

Example: [Bookmark This Page](#)

BlinkList

`<a`

`href="http://www.BlinkList.com/index.php?Action=Blink/addblink.php&url=http://www.BlackBox.com/&Title=BlackBox.com">`
`to BlinkList`

How To Guides for the WorldWide Web

Example: [Add to BlinkList](#)

Del.icio.us

`Add to Del.icio.us`

Example: [Add to Del.icio.us](#)

Digg

[Providing a link to your article that allows users to submit to Digg](#)

`<a href="http://Digg.com/submit?phase=2&url=http://www.BlackBox.com/&title=Black%20Box&bodytext=Black%20Box" to Digg`

Example: [Add to Digg](#)

Furl

`Add to Furl`

Example: [Add to Furl](#)

Newsvine

[Putting a "Seed Newsvine" Link on Your Site](#)

`Add to Newsvine`

Example: [Add to Newsvine](#)

Reddit

[How to make "Submit to Reddit" links](#)

`Add to Reddit`

Example: [Add to Reddit](#)

Technorati

`Add to Technorati`

Example: [Add to Technorati](#)

How to integrate a web site with various blogging sites

`Blog about `

Example: [Blog about](#)

How to integrate web site feeds with various [RSS](#) feed readers

`Subscribe to `

Example: [Subscribe to](#)

This is an automatically-generated printer-friendly PDF version of the HTML web pages at [Bookmark This](#) ← CLICK HERE to get the standard HTML version of this document.

