

MonthlyPLRPackage.com

Newbie Webmasters Guide

Legal Notice:

The author and publisher of this Ebook is Resale Enterprises and the accompanying materials have used their best efforts in preparing this Ebook. The author and publisher make no representation or warranties with respect to the accuracy, applicability, fitness, or completeness of the contents of this Ebook. The information contained in this Ebook is strictly for educational purposes. Therefore, if you wish to apply ideas contained in this Ebook, you are taking full responsibility for your actions.

The author and publisher disclaim any warranties (express or implied), merchantability, or fitness for any particular purpose. The author and publisher shall in no event be held liable to any party for any direct, indirect, punitive, special, incidental or other consequential damages arising directly or indirectly from any use of this material, which is provided "as is", and without warranties.

As always, the advice of a competent legal, tax, accounting or other professional should be sought. The author and publisher do not warrant the performance, effectiveness or applicability of any sites listed or linked to in this Ebook. All links are for information purposes only and are not warranted for content, accuracy or any other implied or explicit purpose.

Table of Contents:

● Finding a Registrar for Your New Website	- 3
● Quick Hosting Setup Guide	- 5
● Websites & Weblogs (Blogs), What's the Difference?	- 7
● What Do You Want Your Website to Do?	- 9
● Hiring Professionals - 5 Things to Look for	- 11
● Working with Templates	- 13
● Image Formats: GIF, JPEG, PNG and More	- 17
● Avoiding the Nuts and Bolts: Content Management Software	- 23
● NVU Free HTML Editor	- 25
● Dreamweaver: The Professional Touch	- 26
● Why Doing It Yourself is Best	- 28
● Definitions of Note	- 29
● HTML for Beginners	- 31

Finding a Registrar for Your New Website

Once you've made a list of domain URL's you're happy with, the next step is finding a registrar. There are two approaches to this. The easiest option is to get your domain name with your web hosting. You often pay a small premium when doing this, but it saves you hassle.

Alternatively you can get your domain name from one of the specialist services:

- **HostGator** - Host multiple websites on one account for very reasonable prices.
- **Hosting Excellence** - - Super fast servers, always up to date software and hardware.

The Confusing World of Web Hosting: Making Your Decision.

Before you can get a website up and running, you need to have a place to put it. Paying for web hosting is, basically, like renting a small amount of space on someone's server and paying what it costs them to send your web pages to your customers. Fortunately for you, though, web hosting has never been cheaper.

Domains and Hosting Together?

Many of the low cost domain name companies have taken to offering you hosting when you buy your domain from them. This is generally an expensive option, and a bad idea - you'll be getting few features compared to buying elsewhere.

So Where Should I Start?

Well, that all depends on what your website is going to need. How many visitors do you expect to have? Are you going to have lots of large graphics on the site? Do you have a lot of articles or products that you want to put in a database? Do you want to have an email address at your website (yourname@yourdomain.com)? On and on it goes. Each host you look at will offer you different combinations of features at different price points, and finding the one that's right for you can be quite a task. Here's a technical-to-English guide to what you should be looking for.

MB (Megabyte) storage. The more MB of storage you have, the more you can put on your website. For most websites, this number can be really very small without it being much of a concern - the pages would be too big for anyone to download and see before they'd be too big to store. You only really need to worry if you're planning to put something apart from plain pages on your site. If you want to make a gallery for your digital photos or let people download ebooks from you, for example, this number needs to be higher.

GB (Gigabyte) bandwidth per month. This is a limit on how much data your website can transfer each month. For small websites, you don't need to worry too much, but as you get more visitors the amount you need will increase sharply, especially if each one looks at lots of pages or downloads large files from the site. The amount of bandwidth your site needs is generally considered to be the deciding factor in how 'big' it is, and how much it will cost you.

MySQL databases. The number of databases your website will have to store things in. It will make it much easier for you if you have one. It's worth noting that if your host may offer some other kind of SQL instead of MySQL (for example, PostgreSQL). You should usually avoid anything apart from MySQL, unless you know what you're doing. PHP, Perl, ASP, JSP, ColdFusion, Python, Ruby. These are all scripting languages, used to

write your website. You should make sure your host offers the languages that any software you plan to use is written in. If you don't have specific requirements, then you should be fine with just Perl and PHP.

Subdomains. These allow you to split your website into more sections than just 'www' - you might decide, for example, that you would people to be able to go to 'shop.yourdomain.com' and 'news.yourdomain.com' and see pages there. You don't really need these, though, as doing the same thing with subfolders ('www.yourdomain.com/shop') is usually just as effective.

Addon domains. These allow you to host more than one website on the same hosting. Some hosts allow unlimited domains - which means you can host all your websites without paying any extra for hosting. This can make a huge difference if you're planning to run an online business as you will almost certainly want more than one website as your business develops.

FTP accounts. An FTP (File Transfer Protocol) account is what you'll use to upload your website to your host. You'll always get one of these. The only situation when you'll need more is if you want to let someone alter things on your site without giving them the master password.

POP3 accounts. POP stands for 'Post Office Protocol', which is just fancy-speak for email. The more POP3 accounts you get, the more email addresses you can have: useful if you want to have sales@yourdomain.com for new customers and support@yourdomain.com for existing ones, for example.

Summary: Web hosting has dropped in price a lot recently - there's a lot of hosts now offering just about everything you need for under \$10 a month. An example is Hostgator, offering high capacity, the latest technology, unlimited addon domains (to host as many sites as you want) and support through email, LiveChat and telephone - all for under \$10 a month.

Quick Hosting Setup Guide

Once you've chosen your web hosting, you'll often find that you're given a set of passwords and technical details, before being left to more-or-less figure it out on your own. If you haven't started a website before, that can be a daunting experience.

Point Your Domain at Your Host.

The email you received should have contained the addresses of some nameservers. Nameservers look like this: ns1.yourhost.com. If you can't find it, take a look at the help section of your host's website.

Once you know your host's nameserver, go and log in at your domain name registrar's website. They all work differently, but somewhere you should see options to configure your domain. Replace the registrar's default nameservers with your host's nameservers. Try going to your domain by typing www.yourdomain.com into your web browser. If it's working, you should see a page telling you that your configuration was successful. If it doesn't, then you should take a break for a day or two - nameserver changes can still take a while to spread across the whole Internet.

Test Your FTP Account.

The next step is to try uploading a page to your website by FTP. Before you can do that, though, you need an FTP program and a test page. The easiest way to make a test page is to open Notepad and write "this is a test". Save it as index.html. When it comes to the FTP program, you have a lot of choice. There's something for everyone: some good free ones to consider are:

- Cute FTP - <http://www.cuteftp.com/>
- FileZilla - <http://filezilla-project.org/>
- SmartFTP - <http://www.smartftp.com/>

Once you've done that, open the FTP program and ask it to connect to your host's FTP server. This is usually ftp.yourhost.com, although you might also now be able to access it through your own website by using ftp.yourdomain.com. Once you're connected, you should browse through the folders looking for any existing index.html file - it'll usually be in a folder called something like 'public' or 'public_html'. Upload your own index.html over this one, and say 'yes' when you're asked if you want to overwrite it.

Now, go to your website in a web browser. If everything's worked the way it should, then you'll see what you wrote in that file right there on your website! You can get started straightaway writing real content to replace that little bit of text - it's always exciting when you realize that your site is out there and ready on the web right now. If you don't see the text, on the other hand, then you might want to refer to your host's support pages.

Set Up an Email Address.

Almost all web hosts allow you to configure your account using a program called cPanel. The host your email sent you should tell you how to access it: it'll usually be something like <http://www.yourdomain.com/cpanel>. If you know the address but you can't get to the page, you might need to disable any firewall software you have running on your computer.

If you've got the cPanel address right, you'll be asked for your username and password, and then you'll be presented with a screen full of icons. Which icons you have will depend on which features you got with your web hosting. Look for the icon called 'email', and then create any accounts you want there.

To check your email, you need to add an account in your email program. This shouldn't be too much trouble: look for an option called 'Accounts' in your email program's 'Tools' menu, and then tell it you want to add an email account. You'll be asked for POP3 and SMTP servers (your host can provide these), as well as the email address and password you just configured in cPanel. Try sending an email to your new address from one of your other accounts, to see if it works.

Other Things to Do with cPanel.

It varies from website to website exactly what you might need to do with cPanel. It makes it easy, though, to do whatever you might need to do, whether it's adding new FTP accounts or creating databases. Don't worry: cPanel is designed to stop you from messing anything up, so it's fine to experiment with it a little. Try out Fantastico®, you will love it.



Websites and Weblogs: What's the Difference?

More and more, people don't have traditional websites: static things where pages can be added, updated or taken away. Instead, they write new material for their website when they feel like it, and then put it up on one page, with the most recent writing first. These people are running weblogs.

How Did Weblogs Start?

Many people say that there have been weblogs (or blogs, as they're sometimes called) for as long as there has been a web. Back when there were only a few thousand websites, the 'What's New' page that announced each new one (yes, there really was such a thing!) worked in just the same way as blogs do today.

Early weblogs included Scripting News, Robot Wisdom and Camworld, which all started in 1997. To begin with, blogs mostly consisted of often-updated lists of useful and amusing links to other websites, but it gradually became clear that the format was just as good for distributing longer articles. Blog software started to be developed, and their popularity quickly exploded. By 1999, everyone was talking about blogs.

Why are Blogs So Popular?

In recent years, the blog format has very much taken over from the 'personal home page'. People seem to find it much easier to just put a kind of public diary online, instead of putting up a little biography of themselves and a collection of articles. It's more personal, more fun, and more interactive day-to-day.

Businesses have started to open blogs too - in many ways, they're like a replacement for newsletters. A regularly-updated blog gives customers a great sense of what a business is like, while giving the business a great way to keep communicating with its customers and being useful to them, even when they're not buying anything right this minute. In my opinion, the biggest reason for blogs' popularity is that they take publishing to the web very easy. You don't really have to know anything about what's happening behind

the scenes: blogs finally make publishing your thoughts for everyone to see as easy as posting to a forum or sending an email. In a way, blogs fulfil the original promise of the web.

Weblog Software.

Today, there's a lot of blog software out there - if you want a blog, you're spoiled for choice. What you get will depend on how comfortable you are with technical stuff, and whether you want it to be part of your main website or not. Movable Type. This is software that you install on your web server. You simply log in and type your post, and it creates your pages for you. Movable Type can be a little complex to

set up, but you can use a version called Typepad that is hosted by its creators instead of using your server. Blogger. You don't install Blogger on your server - instead, you give it your FTP password and let it upload files to your web server for you.

If you don't have any hosting, you can also host blogs for free at Blogger's Blogspot. Blogger is owned by Google. <https://www.blogger.com/start>

WordPress. WordPress is a free alternative to blogging software. It works in basically the same way as Movable Type, but without the restrictive licensing and with nicer-looking default templates. Many

people have switched to WordPress out of frustration with Movable Type and not looked back. You have to host it on your own server, but it's very simple to set up - don't be scared! <http://wordpress.org/>

LiveJournal. LiveJournal is a completely online service, meaning that it has nothing to do with your website, except that you can link to your LiveJournal if you want. LiveJournal is more social than most blogging, allowing you to join communities relating to your interest. <http://www.livejournal.com/>

There are plenty of other online services, but they're all pretty much the same: MSN Spaces, AOL Journals, and so on. You're unlikely to get taken very seriously if you have a blog at any of these places, although it'd be easy. In the end, it's all about power versus convenience: the more work you put in to get your blog working, the more likely that it's going to be what you really wanted it to be. If you're creating a website anyway, you'd be silly not to put a blog on it.



What Do You Want Your Website to Do?

There are a lot of different kinds of websites - and there are a lot of people who know they want a website, but aren't even sure why. If you don't already know, you need to figure out exactly what it is that you want your website to do. What kind of thing do you want to put on it? Who are you trying to get to visit? Here's a quick guide to some of the most popular website types.

The Business Card Website.

For many people, having a website is something they feel the need to do as a duty, not because they really want to do it. In this case, it's perfectly justified to put up a website with nothing more than your name, your business name (if any), and your email address. You could also include your real-world address and your phone number, if you're comfortable with that. The key here is to at least put up something for someone who happens to be looking for you with a search engine. You have to understand, though, that these kinds of sites can be as frustrating as they are helpful. Friends and co-workers may be excited to find your site, only to say "oh... is that all it does?" Customers, especially, have a tendency to get upset, especially if they just wanted to know something about your products or pricing. Not having a complete website makes work for them when they have to ask you about these things, and for you when you have to answer.

The CV/Portfolio Website.

An extension of the business card format is to make your website brief, and directed at getting people to call you if they're interested in you. It could just be a listing of the jobs you've had, or it could be an archive of the articles you've written for various publications - whatever, it's there to get you work. These kinds of websites can be effective, although you should realize that you're more likely to get some work because of a site that people find useful than because of one that's just about you.

The Brochure Website.

They're often-mocked in web design, but they're not really that bad: brochure-style websites simply reproduce the kind of material you would send out in a brochure to an interested customer, complete with pictures and technical details. Instead of actually doing any business online, you give the customer your contact information. In many industries, really, this is the only way to work. There are things you just can't do with shopping carts and credit card processing, after all, especially when it comes to services. These kinds of websites are especially prevalent among businesses with a more local focus.

The E-Commerce Website.

There are two situations where e-commerce works really well: if you're selling a service that can be provided over the web, or if you're selling a small product that you can handle shipping for. In these situations, e-commerce websites are very powerful. You can sell your product direct online, cutting out whatever middlemen there might be.

The Content Website.

Ever more popular, content websites work by providing useful writing and then putting ads around it. That's the whole business model in a nutshell, but it works surprisingly well. The more writing you can do (or get), the better it works for you. If you want, you can even leave off the ads and put up a small button asking people to donate to your site if they found your information helpful - people can be nicer than you'd expect.

The Community Website.

Finally, an often-ignored kind of website is one that exists to serve a community. The community might be geographical (people who live in the same town), or it could be an interest or hobby community (a forum for people who really like a certain genre of TV show, for example). Community websites can be fun to participate in, and they can make a lot of money if you can find advertisers who are interested in the very specific audience your site has got. If your site gets popular, you might even be able to charge people for membership! Even if it stays free and small, though, it's nice to have a web community to call your own, and you're likely to benefit in all sorts of ways that you wouldn't expect



Hiring Professionals: 5 Things to Look For

So you've decided that it might be best to leave web design to the professionals, have you? Well, the unfortunate thing about web design professionals is that not all of them are exactly professional. Some of them are working from wildly out of date knowledge, and, well, some of them are working from their parents' basement.

How can you protect yourself when you decide to hire a professional? Well, here are some guidelines. Qualifications are Meaningless.

Web design is a new enough industry that what qualifications there are have ended up being largely stupid and meaningless. I've never met a good designer who has a qualification in web design - a qualification in proper graphical design is, of course, something else entirely. You can guarantee that most people with web design qualifications did them at a community college to try and get rich during the dot-com boom. These people know nothing about web design. What you should pay attention to when you're looking at web designers is which of their skills they think are important. If they're marketing their Flash expertise to you, you probably don't want to hire them. If they can competently explain what XHTML and CSS are and why they're good for your website, then they're a better candidate.

1) The Portfolio.

The easiest way to tell whether the designer you're considering is any good or not is to take a look at their portfolio. Good designers will have a diverse portfolio, with plenty of attractive sites that they've built for all sorts of customers, and they'll be able to explain to you why they built each site the way they did. If the designer doesn't have a substantial portfolio yet, you shouldn't write them off there and then - but do say that you're going to need to see something before you commit yourself. A good designer will happily do a first-draft design for one of your pages just to show themselves off. If you're happy with what they've done, then the chances are that they've got the rest of the design in their head, ready to code up and send to you.

2) Usability is Vital.

You should make sure that the designer you're talking to realizes that they're building your website for users - all users, no matter how they want to use it or which web browser they have. Amateur web designers have a tendency to come up with things that they think are impressive-looking, either visually or technically, while ignoring who the website is actually for - your visitors. Some questions to ask the designer include: Which browsers are you going to test with? (They should reply with 'as many as possible'). What is the purpose of my website? (They should be able to explain the point of the site simply).

3) Setting a Price.

Many web designers simply have no idea what kind of price to charge for their services. If you post a project to a freelance site where designers can bid on it, you'll get bids ranging from \$50 to \$5,000. Both ends are pretty obviously ridiculous. At one end, you've got kids with no experience and Indians who don't speak English; at the other, you're paying pointless committees to have meetings for a week.

4) Independent Pro.

Your best bet is to find someone professional but independent, experienced but not corporate, and with a price that reflects the quality of their portfolio. As with all things in life, though, don't be afraid to negotiate - and never agree to let them do however much work it takes and pay by the hour.

5) Hosting Offers.

Finally, beware of web designers who are trying to sell you a whole package, together with a domain name, web hosting, and the rest. Operations like this are almost always flawed in one area or the other, since designers aren't really very technical people on the whole. They'll either be web hosting specialists who think design is so easy anyone can do it, or they'll be designers who are reselling some cheap hosting package for a premium. On the web, it's better to buy hosting yourself.



Working with Templates

If you don't want to deal with designers but you don't want to design a website yourself either, there are plenty of websites that would just love to meet you. They sell templates, which are an easy way for anyone to buy an already-existing design and apply it to their website.

Free Templates.

A quick search for 'free website templates' turns up a lot of offers. You have to realize, though, that almost any template you get for free is either going to be really very amateurish - looking through the results, in fact, a lot of it is downright nasty. There will also probably be some annoying conditions of use, such as having to link back to the designer's web page from your page, or not being able to use the design commercially.

If you're really determined to get a template for free, a better way of doing it is to use the ones that come for free with whatever software you're using. WordPress, for example, has some very clean, attractive templates. However you do it, though, you have to realize that free templates will never be unique: your website will look just like hundreds of other websites out there that found the same free template you did.

Cheap Templates.

Once you start to get into the territory of paid-for templates, things start to look up. The idea behind sites that sell templates (templatemonster.com is the market-leader) is that they can pay designers a proper rate to do something good once, and then resell it as many times as they want for a relatively low price. This lets designers be paid for as many designs as they want without ever having to deal with customers, and it lets customers buy and use the designs for a much lower price than they'd usually pay for something a lot worse - and that they can't see in advance.

Although the effect isn't quite as bad as with free templates, you still have to understand that other people out there are going to have the same design as you. This can be a problem, especially if you use one of the big template sites and pick the obvious design for the type of site you're running. Used carefully, though, it can be one of the best ways to get a good-looking site up without breaking the bank.

Exclusive Templates.

After realizing how much some people dislike other sites being able to use their design, many template sites started selling exclusive templates - ones that are only sold once, to one website, and then taken down. They found themselves with a runaway hit on their hands. The reasons for this might not be immediately obvious, as buying exclusive rights to a template are often more expensive than just paying a designer to do it to begin with, and you get less say in what the site looks like. If you ask me, the popularity of exclusive templates is down to removing the often fraught relations between a web designer and the customer.

Customers all too often come into the design process with all sorts of requirements and preconceptions, and designers will fail to understand what customers care about and what they don't. This way, designers are free to create something great, and customers can take it if they like it or look at hundreds of alternatives if they don't. It takes all the uncertainty and negotiation out of both ends, and leaves both the designer and the customer much happier than they would have been. Of course, if there is something small you want changed, most sites are happy to get the designer to do it for you for a small extra fee.

Adding Graphics to Templates.

Some people get very intimidated when they here the Graphic Design. It is true, you do need a bit of talent to create some good graphics for your site, you should a hire a professional only when you really need them for extra special graphics. Otherwise you can try out our sister site at <http://resalegraphics.com> for thousands of ready to go images like "Order Buttons", "Web Page Headers and Footers", "Check Marks", and much more. All free of charge.

Putting Text in Templates.

Once you've got a template, the only remaining step is to take your text and put it into the template. The designer might be able to help you with this, or you might prefer to add the template to whatever software you plan to use so that your content and navigation can be added automatically.

Building a Budget Website.

Once upon a time, building a website was very expensive. Now, though, you can have a site up and running for the price of a used book, if you're frugal and careful. Getting the cheapest website you can is a great first step on the ladder, to get started on the web and see if it's for you.

Five-Dollar Domains.

If you're willing to take the less popular domains like .info, or some of the ones for specific countries, then you should easily be able to get your hands on a domain for less than \$5. Some countries, such as .tk (the small island of Tokelau), even offer their domains for nothing!

Free Software.

Nowadays, it's easily possible to build a website using nothing but software you can get for free - most of the best scripting languages are free, and each one has had a lot of free software built for it by hobbyists. Check out <http://sourceforge.net>, which is a big repository of free software.

We recommend the following:

Free PDF Creation Software - Now that you have edited the eBooks source file, you will need to convert it into PDF (Adobe) Format. Here is a link to 100% Free PDF Creation software that you can use to make a professional ebook. <http://www.pdf995.com/>

"GIMP" - Free Alternative Software to Adobe Photoshop ® - Would you like to be able to edit, change, or create your Photoshop compatible graphics? This is a 100% free alternative that you can use to do just that. <http://www.gimp.org/downloads/>

OpenOffice.org - Get a 100% free version of an office suite that will work with commonly used Microsoft ® Office © Suite of software programs. <http://www.openoffice.org>

Free Search Engine Optimization Software - That's rights FREE. There is a 100% free version of this great software and it never expires. <http://tinyurl.com/3exm4y>

WinZip ® (Zip utility Software) - All of our delivered via an email sent to you. Inside that email is download links to your purchased files. Those files are delivered in a Zipped folder. You will need a zip file extractor to access your products once you have downloaded them <http://www.winzip.com>

Free Adobe Reader Update ® - Most of the eBooks that we sell are in PDF format, which requires Adobe Reader, to open and read. Most computers come with it already installed, but if for some reason you do not have it, get the free version here. <http://www.adobe.com/products/acrobat/readstep2.html>

Free Link Popularity Checker - It is very important to know the link popularity of your website, as well as, your competition. Use this 100% free software to track your websites link popularity on all of the major search engines.. Get it Here. <http://www.master-resale-rights.com/LPC.zip>

You might think that free software would be less functional than paid-for software, but you'd be wrong. Plenty of free software is simply implementations of standard software, and it works perfectly well - if you want a forum, for example, there's no clear advantage in paying massive license fees to vBulletin (the biggest seller of forum software) instead of just installing the free phpBB. The free software gives you more flexibility, and yet comes at no cost.

Free software has become an ideological movement, for people who want to be able to modify their own software, and much of the free software out there is quickly becoming widely-used and standard. Using free software doesn't make you look cheap, because users are used to seeing it everywhere - even better, the chances are that they already know how to use it.

Templates.

Depending on what kind of website you're running, you could use the design templates that come with your free software - they're usually perfectly adequate. If you don't want to do that, then a quick look around at a site like [templatemonster.com](http://www.templatemonster.com) is sure to turn up something suitable for your website that only costs a few dollars.

Pay as You Go Hosting.

Instead of asking you to pay monthly for hosting, more and more hosts are starting to offer 'pay as you go' hosting, which means that you only pay for what you use. This saves you a lot of money, because websites that are starting out rarely use all the features and bandwidth they're paying their host to provide. At nearlyfreespeech.net, for example, you add money to an account and then pay one dollar for each gigabyte of bandwidth you use. Most of these hosts allow you to start an account with very little money - the minimum is usually \$5. If you keep your site light on graphics, that first \$5 can last you a very long time.

Guerrilla Marketing.

Finally, one of the biggest costs associated with any website is marketing it - whether you're planning to pay for banner ads or ads in search engines, marketing is a big cost. You can save money, though, by resorting to more 'guerrilla' techniques, such as becoming involved in online communities than you think might be interested in your website. The biggest free marketing technique out there is SEO (search engine optimization), which is when you build your website in a way that makes it more attractive to search engines, getting you targeted visitors for free.

Taking it Further.

Once you've built your budget website, do you need to upgrade it later on when you start to get lots of visitors? Often, the answer is no. You might wish to buy a more prestigious .com domain name, and you might want to pay a professional designer to improve your design, but in most cases the path from a budget website to a big one isn't all that costly either. You might think you're building a website 'on the cheap' but, really, that's the most sensible way to do it now - while you can go and spend thousands of dollars on software and hosting, you're unlikely to see any real benefits at all.

There's More than One Web Browser.

If you're like 90% of web users, then you use Internet Explorer, Microsoft's web browser. Why? Well, because it comes with Windows, usually, and it's there on your desktop when you first want to use the

web. When you're creating a website, however, you have to consider the other 10% of the web's users - the ones who use alternative web browsers.

If you don't test your site in each one of these browsers, you might be in for a nasty surprise when a large part of the web can't use it. Here's a guide to the most common alternative web browsers.

Mozilla. <http://www.mozilla.org/>

While Mozilla is a web browser itself, it is also an engine (Gecko) that powers a lot of other web browsers. The Gecko code is free and open-source, created by Netscape, and is currently Internet Explorer's biggest competitor. It works on Windows, Mac, Linux, and almost everything else out there. The advantage of Gecko is that if you've tested with one of the Mozilla browsers, your site should work on all of them. The Mozilla browsers include Mozilla Firefox, Netscape, Camino, Kmeleon, and lots of browsers for the Linux operating system. You'll probably find it easiest to download Mozilla Firefox from getfirefox.com and test your website using that.

Opera. <http://www.opera.com/>

Opera is, in many ways, the alternative browser, for people who are too odd to even want to use a Mozilla browser. It's very much a niche product, developed by a small Norwegian company, and many more technical users like it because of its constant innovation when it comes to features - anything you like in another web browser was probably available in Opera first. Opera is for Windows, Mac and Linux. You can download a free version of Opera from opera.com. It has ads, but it's perfectly fine for testing. If you like Opera enough to actually use it for your own web browsing, you can pay a one-time fee of around \$40 to remove the ads.

Safari. <http://www.apple.com/safari/>

Safari is now the official Mac web browser, which means that it's important to test on it if you want Macintosh users to be able to see your website. Unfortunately, Safari doesn't run at all on Windows. However, Safari uses the same engine as a browser called Konqueror, which can run on PCs on the Linux operating system. Linux is free, and you can easily download it, burn it to CD, and run it straight from the CD. Knoppix (www.knoppix.com) is a popular and easy to use kind of Linux for this purpose.

Alternatively, if you don't want to mess around with a whole other operating system, you could try a service like BrowserCam (www.browsercam.com). They will load your website into many different browsers, and then send you pictures of it to let you see if there are any problems that need to be fixed. Because of the bandwidth and the number of computers that have to be involved, though, most of these services aren't free.

Lynx. <http://lynx.browser.org>

Lynx works on a lot of very esoteric operating systems, but works fine on Windows too. It's a text-only browser, and it pays no attention to layout or graphics.

Why would anyone want that, you wonder? Well, Lynx is mostly popular among blind people who use screen-readers to turn web pages into speech. How well your web page works in Lynx is often considered to be a test of how accessible it is to anyone with disabilities, as well as to anyone who turns off things like JavaScript in their browser settings.

Image Formats: GIF, JPEG, PNG and More

When you want to put graphics on your website, you'll face an unexpected problem: what format should they be in? On their own computers, many people save pictures in Windows' default BMP (bitmap) format, but the files it creates are simply much too large to put on a website - they'd take about a minute for visitors to download and use up all your bandwidth in the process.

When you put pictures on the web, you need to consider the trade-off you want between image quality and speed: the smaller the file, the worse it's going to look. To help you out, here's a comparison.

GIF

GIF stands for Graphics Interchange Format, and was the first image format used on the web. It was invented by CompuServe in 1987, updated in 1989, and hasn't changed since - and it shows. Images stored in GIF format can only use a maximum of 256 colors, which makes things like photographs and logos look terrible. GIFs popularity is mainly due to it being first and producing very small files, although it is also notable for being the only image format that allows you to create small animations.

Really, the only things you should use GIFs for now are files that have a limited number of colors, and are technical in nature - diagrams, for example, work well in GIF format. Things like photographs that use many colors will come out looking very strange.

JPEG

JPEG was designed by the Joint Photographic Experts Group, who gave it its name. It was designed as a format suitable for storing high-quality photographs at low file sizes - file sizes small enough to put on the web. Today, the format is supported in almost all web browsers, and is also the format that you'll get photos in if you take them with a digital camera.

The most important feature of JPEG is both the best and worst thing about it: lossy compression. The word 'lossy' means that data is lost from the picture when it is saved at smaller file sizes. Image-editing programs will generally let you choose how much compression you want, from none (highest quality, large file size) to 100% (very small files, but terrible quality). Unfortunately, JPEGs that have been compressed too much come out looking worse than useless, but many people still use high compression settings out of a misguided desire to have the very smallest files possible. If you've ever seen images on the web that look very 'blocky', you've been a victim of JPEG compression. If you do use JPEG, then, it's really recommended that you turn compression off altogether, or use a maximum of about 25% - the files are quite small already, without going overboard with the compression.

PNG

PNG stands for Portable Network Graphics, and is the newest web image format in widespread use. It was designed as a replacement for the outdated GIF format, allowing files to use millions of colors instead of only 256. PNG files have smaller file sizes than GIFs, although they are often larger than JPEGs, since PNG compression is lossless (never loses any image quality).

The most useful feature of PNG is that it supports something known as 'alpha transparency': basically, images with transparent backgrounds that blend in perfectly. The only thing stopping widespread adoption of this feature is that it isn't currently supported by Internet Explorer, but there is a workaround for this problem: search for 'AlphaImageLoader' for more information.

Converting Between Formats.

For most purposes on the web, all the graphics you want to use should either be in JPEG format (for photos) or PNG format (for less complicated graphics). That's a problem if you've got a collection of images in all sorts of other formats.

Luckily, a good image editor should be able to convert from any format to any other very easily. In Paint Shop Pro, for example, you simply open your images and save them again using whatever format you want - you can even run the 'Batch Converter', which will convert a whole folder full of files all at once. If you don't have an image editing program, there are plenty of free image viewers that will do the same job for you.



The Many Flavors of HTML

HTML (Hypertext Markup Language) is the language of the web - every website out there is written in some kind of HTML. Because of the rapid evolution of the web, though, HTML grew quickly in a very unplanned way, which can lead to problems if you're not sure what kind or version of HTML you're using. Here's a quick history of HTML's flavors so far.

A Long, Long Time Ago...

The first version of HTML was created by the web's inventor, Tim Berners-Lee, and was loosely based on an existing standard called SGML (Standardized General Markup Language). This very first version didn't have an `img` tag, which meant that no graphics at all could appear on web pages. Berners-Lee informally extended the language, but didn't standardize it.

As the web grew, the lack of standardization started to make it difficult for web browsers to interact - one web browser might have a new tag that others didn't support, meaning that people would see pages completely differently depending on which browser they used. In 1995, HTML was formalized as a standard named HTML 2, which was the version that the first mass-market web browsers were based on.

As they extended the standard further, an HTML 3 was introduced in 1997 to keep up-to date.

HTML 4 was introduced later that year as an effort to clean up the standard, making it clear that some tags should no longer be used. Apart from a few minor fixes in 1999, this is the version of HTML that is still in use today.

DHTML

Parallel to this development, though, other languages were being developed that could be included in HTML documents: languages like JavaScript (for interactive pages) and CSS (for styling). DHTML (Dynamic HTML) was the name given to the combination of HTML and these technologies. To put it simply, HTML is for web pages while DHTML is for 'web applications'. As people start to do more and more things on the web that they used to do with separate programs, DHTML techniques are becoming ever-more popular.

XHTML

Sometimes considered 'next-generation HTML', XHTML is a stricter version of HTML that makes it follow XML standards. XML (extensible Markup Language) is a standard for HTML-like languages that is being used for more and more purposes, including configuration and sharing data.

Stripped of the technical talk, XHTML can basically be thought of as a stricter version of HTML. Where HTML is often messy and hard to test, XHTML is strictly standardized and can be run through automatic 'validators' that will point out any errors you've made. This improves cross-browser compatibility and makes web pages much easier to maintain, since it mostly forces information on the style of the page to be separated from the actual text of the page.

XHTML exists in a few different versions: there is a 'transitional' version, which lets you keep using some old practices from HTML4, and there is a 'strict' version, which is the one you need to use to get most of XHTML's benefits. The web's standards body, the W3C, runs an HTML validator at <http://validator.w3.org>.

What Does All This Mean to Me?

You might be wondering at this point why exactly you need to know about the different kinds of HTML. Well, as ever, the answer is that you need to choose one before you start developing your website. You have to be aware of which versions your tools support to know whether your tools can work together, and you should aim to pick the kind of HTML that will be most suitable for your site.

At the moment, XHTML is recommended for most websites, simply because it makes the whole process much easier, especially if you use an editor that saves to XHTML automatically. The only situation in which you should really keep using HTML4/DHTML is if you're designing a web application instead of a web page. If your site is, like 99% of the sites on the web, designed to give information more than it is designed to do anything else, then you should be using XHTML, preferably the strict version.

Clean Page Structure: Headings and Lists

When HTML started, people put all sorts of things on their pages: there was a tag to say which font you wanted your text to be in, a tag to say you wanted it to be in the center of the page, and so on. Now, though, that way of writing pages is out-of-date and very inefficient compared to keeping content and style separate using CSS.

What is CSS?

CSS stands for Cascading Style Sheets - it is basically a way of saying once what you want your pages to look like, instead of having to repeat it in HTML all the way through the document. In old style HTML, for example, this kind of code was a relatively common sight:

```
<p><font face="Arial">Welcome to my website!</font></p>
<p><font face="Arial">I hope you enjoy your visit.</font></p>
Now, though, you can remove the font tags altogether, and just have this:
<p>Welcome to my website!</p>
<p>I hope you enjoy your visit.</p>
```

At the top of your website, you put a style tag, like this:

```
<style>p { font-family: Arial, }</style>
```

Now, instead of having to say again which font you're using with every new paragraph, you've told the browser that you want every paragraph you've got to be in Arial.

Headings and Lists

Thanks to CSS, you can make documents that are more 'semantically correct' - that is, they would make sense to a human reading them, instead of having to be weighed down with lots of extra presentation code. This has two great effects: it makes web pages smaller (and so faster to download), and it makes them simpler.

On a modern web page, the only things you should ever need to include apart from paragraphs of text are headings and lists. After all, web pages are just text, graphics and navigation put together in a particular order - there's no reason for things to be done as messily as they often are.

You use headings for the title and subtitles of your page - they're the HTML tags that begin with h. You might, for example, write <h1>website title</h1><h2>article title</h2>.

Lists, on the other hand, can be used for pretty much anything else that isn't paragraphs of text. Instead of just putting links one after the other to make a navigation bar, for example, you should put them in a list, using the `ul` and `li` tags. Not only is this easier for you to read and add to, but it's also more compatible with non-graphical browsers.

A typical list looks like this:

```
<ul><li>item 1</li><li>item 2</li></ul>
```

Bold, Italic, and Images

Of course, in practice, you'll need a few more tags. CSS lacks any good way of making individual words bold or italicized, so you can still use your `b` and `i` tags. Images, of course, still need a tag of their own too, although you might consider putting your images in a list if you have more than one.

In theory, at least, that still means that it should be perfectly possible to create a clean looking web page using only six tags: `h`, `ul`, `li`, `b`, `i` and `img`. And, yes, it is very possible - if you can stick to this attitude towards web pages, your page will be extremely clean, quick to download and fast to display. If you've ever sat and waited more than a few seconds for a page to load, you'll know how important this is.

Custom Stylesheets

One more advantage of this approach is that it lets your users view your website however they want to. There are a surprising number of users out there who are elderly or just want some consistency on the web, and they have their own CSS Stylesheets to add to pages to make text bigger, for example, or make the layout simpler.

Once you've written your page cleanly, you can even offer visitors a choice of Stylesheets yourself - you can write more than one and then offer an option to switch between them. This makes redesigning your page much easier, if you ever need to, since you can simply swap one set of CSS for another and even leave the old one available for any visitors that prefer it.

The Importance of Validation

Once you've written a web page, you can upload it to an HTML validator. This site, run by the web's standards body, will check that your site is valid ('correct') HTML, and give you some idea of how to fix it if it isn't. This is an essential step in the development of any website - as vital as running your text through the spell checker - but whenever I recommend it there's always someone who wonders why it's so important. Well, here's why.

You Know Your Code is Correct

If your code validates, then it's correct, and therefore very likely to work as intended on every web browser out there. If you don't validate your pages, then you might find that people who visit your site with less forgiving browsers see nothing at all. Correct code is more likely to display correctly on many different browsers, because it puts them into their 'standards' mode. If code is even slightly incorrect, many browsers will use a different way of displaying it, known as quirks mode, which is designed to handle old and bad HTML, takes a long time and may make your page end up with errors you didn't expect.

Without web standards, you end up going back to the bad old days of having to develop entirely separate web pages for different browsers. Validating by the standards ensures that all working browsers can view your content - if they can't, the fault's with them, not with you.

Search Engines Like Valid Pages

When it comes time for a search engine to add your page to its results, it's going to have a much easier time understanding the page if it's been validated. This will often get you a higher ranking in the results, which means free visitors for you. If your page isn't valid, search engines will often miss keywords in your pages or not understand your navigation, and may list nonsensical parts of your code under your site's name in the search results - not exactly helpful to potential visitors who want to know what your site is about.

Mobile Devices

More and more people are accessing the web using mobile devices like mobile phones and PDAs, and these devices have a lot of trouble with code that isn't valid. Because they have limited processing power, it would take them a very long time to try to untangle invalid code - they will simply strip out the formatting and do the best they can with it. Writing valid HTML lets users with mobile devices see your pages as you intended.

People with Disabilities

When you write valid code, it becomes much easier to view with things that aren't web browsers, such as screen readers. Technology for disabled people doesn't tend to be as forgiving as web browsers, so having valid code is important when it comes to working with these programs.

Future-Proofing

Before your code will validate, you need to explicitly say which version of HTML you had in mind when you created it. This future-proofs your code, as each version of the standard doesn't change once it's been decided on: a valid XHTML 1.1 page will always be a valid XHTML 1.1 page, even if everyone else has moved on to XHTML 5. Once you've validated your site once, you can put it on the web and be confident that people are going to be able to read it for a long time to come.

Finding Errors

If there's a mistake in your website's code, validation gives you an easy way to track it down and fix it. Before validation, people had to test their site after each change and look carefully to make sure that nothing had gone wrong. Writing valid code lets you use programs that will examine what you've written and point at the exact place where the code doesn't validate.

A List of Validators

Here are a few validators that you can try. Most HTML validators are online, but there are a few that you can download and use on your own computer.

- The W3C validator: <http://validator.w3.org>
- The WDG validator: www.htmlhelp.com/tools/validator
- CSE validator: www.htmlvalidator.com (downloadable)
- WebTechs validator: www.webtechs.com/html-val-src
- Doctor HTML: www.doctor-html.com (downloadable)

You might also be interested in visiting the W3C's main site at www.w3c.org, as well as the Web Standards Project at www.webstandards.org.

Avoiding the Nuts and Bolts: Content Management Software

If all this talk of coding and designing scares you off, you might want to know that there is an alternative to all this. You can install a kind of software called a Content Management System (CMS) that allows you to put content up on the web without ever knowing a thing about HTML.

Fantastico

Depending on your host, you might find that you already have a selection of CMSes available and ready to install from your cPanel. Log in, and take a look for the Fantastico script installer. If you have it, then you'll be able to read a description of each piece of software you have available to you - try out a few of the CMSes to see which ones you like.

Finding a CMS.

If you don't have Fantastico on your host, or you don't like what it offers you, don't worry: there's plenty of choice out there on the web when it comes to CMS software. For finding free CMS software, a truly excellent resource is www.opensourcecms.com. At that site, you can use the menu at the side of the page to see lots of open source CMSes running before you commit to downloading them and installing them yourself.

Textpattern, Drupal and Plone are very popular right now, so they're a good place to start. If you're considering commercial CMSes as well, then you should take a look at www.cmsmatrix.org, which provides an up-to-date comparison of almost every CMS out there. Be prepared, though, that commercial CMSes can be ridiculously expensive or unnecessarily expensive.

Getting a Custom CMS Built

If nothing out there seems to meet your needs, you might consider having a web designer build you a custom CMS in a scripting language like Perl or PHP. Any web designer worth their salt should have something basic already that they can build whatever features you want onto. This can be really good when it comes to making your website work the way you want, since the CMS will be built around your website to make it as easy as possible for you to modify.

Using a CMS.

The whole point of a CMS is to make it much easier to add content to your website and to edit the content that's already there. Once you've installed your CMS, you will generally be able to log into its user system using a special admin password. This will add 'edit' options to the existing pages of the site, as well as giving you a 'new page' link somewhere to allow you to create a new page.

When it comes to actually writing the content of the pages, most CMSes will make it easy for you to copy and paste from programs like Word: they shouldn't require any special HTML formatting. Some will require you to mark words with special symbols if you want them to be bold or italic, but it shouldn't be too troublesome.

Changing the design with a CMS usually involves installing a template into a template folder and then selecting it in the options. Creating your own templates can be complicated, depending on what software you're using, but it shouldn't be any trouble for a web designer, and most template sites will provide designs in a format suitable for this kind of use

Finding a Hosted CMS

Once you know which CMS you want, an alternative to installing it on a web host that doesn't necessarily support it is to do a search and find a host that specifically supports your CMS. You shouldn't have too much trouble finding a host somewhere that will support you - if nothing else, you might try opensourcehost.com, which supports almost every open source CMS out there.

Using a host solution might be a little more expensive, but it will save you a lot of time in configuration and a lot of problems if anything goes wrong. Using a hosted CMS is one of the quickest ways to set up a website: you simply pay the host, log in, add your content and you're away.



NVU Free HTML Editor

<http://www.nvu.com>

One way to create web pages from scratch without using HTML is to use an editor that hides the HTML from you, letting you edit a web page as easily as you would use a word processor. These programs are called WYSIWYG (What You See is What You Get) editors.

Using NVU

NVU is very easy to get started with: you can either create a single page, or a whole You can use the buttons on the toolbar to do simple things like set your text's font, make text bold and italic, make links and insert graphics.

Going up to the menus will give you access to a few more complicated functions, such as table creation. Part of the reason FrontPage is so popular is that it has all sorts of little scripts ready to insert into your page, including hit counters and animations.

NVU Templates - <http://www.nvutemplates.com/>

One of NVU's strong points, however, is that it has an easy-to-use templating system. This means that you can download templates and easily use them to create new pages in NVU. It will create a navigation system for you as you go, using information from the template. This can be a quick and easy way to get started on your website, although you'll often need to be careful to avoid doing anything that causes the carefully worked-out layout of the template to break.

Dreamweaver: The Professional Touch

<http://www.adobe.com/products/dreamweaver/>

Dreamweaver is sometimes seen as FrontPage's main competitor but, really, there's not even a comparison to be made. Dreamweaver might be expensive, sure, but there are serious web designers out there using it and getting work done - I can guarantee you that no real designer has ever used a copy of FrontPage to design a website.

Consequently, Dreamweaver is way out in the lead in terms of market share, with about 80% of the users.

So what do you sacrifice to get a better WYSIWYG editor? Well, apart from the higher price tag, you also sacrifice a certain amount of simplicity. Once you get used to it, though, Dreamweaver isn't as hard to use as you might think.

From the People Who Brought You...

Dreamweaver is part of the Macromedia Studio - that's the company that produces the editor and player for Flash, as well as the well-liked Fireworks graphics editor and the Freehand drawing program. Their software is particularly popular with designers, and all it works on the Mac, but it has recently started to be quickly adopted by non-expert users as well.

Dreamweaver Doesn't Mess With Your Code.

If you've already done some of the coding for your site, or you're editing a template or a design that was done for you by someone else, then you don't need to worry about Dreamweaver re-writing all your code and breaking it. Dreamweaver will leave your code alone for the most part, unless you explicitly tell it to alter it. This might not sound like much, but it's really refreshing to see after using other HTML editors.

For this reason, Dreamweaver is often considered to be the very best software out there for working with HTML templates, and most templates that you can buy will be provided to you in Dreamweaver's format. Once you've got them, you can easily open them and insert your content.

Standards

The code Dreamweaver used to produce was quite bad at conforming to standards - it worked fine on most browsers, but it didn't validate. In the most recent versions of Dreamweaver, though, not only does the code Dreamweaver produces validate, but it even has a validator built in. You can pick which standards you want Dreamweaver to code to, and it will stick to them for you, even if you choose the strictest ones out there.

Dreamweaver was one of the first programs to support visual XHTML editing, and has received a lot of credit for it - all you need to do to turn this option on is tick the box marked 'Make Document XHTML Compliant' when you create a new page.

CSS, JavaScript and PHP

A large part of Dreamweaver's power comes in how easy it is to edit not only HTML, but also the things that go with it. Dreamweaver comes with a formidable JavaScript library that does most of the useful things that can be done with JavaScript, and makes it very easy to apply CSS styles to different parts of your page.

In the latest version of Dreamweaver, you can even make interfaces to a MySQL database using PHP without doing any programming at all. This is a very useful feature, and saves a lot of headaches for many people who just want to make a simple database-driven website without learning PHP.

Books about Dreamweaver

Dreamweaver is a complex and useful enough program that whole books have been written about it - and to get the most out of the program, you should really get a good one and read it through.

Why Doing It Yourself is Best

There are two ways that most people who create web pages do it: either they hire a professional web designer, or they use some visual HTML editing software. You could say that hiring a designer is like getting someone to make furniture for you and deliver it, while using an editor is more like buying flat-pack furniture.

What I'm here to tell you, though, is that you should consider doing things another way. I'm telling you that you should go get yourself some wood and a saw. In short, you should do it yourself.

Why Would I Do That?

No matter what you might do, unless you're doing it yourself - that is, coding your website by hand - you're not going to have full control. Doing it yourself gives you control over every tiny, tiny detail, and puts you in a situation where there should never be a problem you can't fix. You're not relying on a company, or a designer, or anyone else - whatever goes wrong, you can fix it.

Thanks to XHTML and CSS, creating websites by hand has never been easier - in fact, once you know the basics, it's often easier than fighting with some piece of software to get it to make things look the way you want. Once you know the tag for an image, it's much easier to just type it than go through a bunch of menus. When you know you want something to be a certain width, it's easier to type that in than to try to make it the right size using the mouse. Web design benefits from the kind of preciseness that you get from doing it by hand.

At every stage in the process, you know what you've done and you know what you have left to do. If something isn't working the way it should, you can easily look through, find the problem, and make it right. Altogether, not only do you save the money that you would have spent on Dreamweaver or FrontPage, but you also end up with a better website in the end.

Tables and Coding

For a long time, the biggest reason to stay away from hand-coding websites was that most complex layouts were done using tables, and tables were hard, especially when you had to put one inside another. Any modern website, though, really shouldn't be using tables at all, which means that you won't have to learn how to do them - that takes down the biggest barrier to hand-coding. HTML is very easy to figure out, and from there all you have to do is learn a little CSS (there are plenty of good books on it, and there's not that much to learn altogether) and you're away.

Re-using Code

There are only a limited number of things that you'd want to do in HTML, and they've all been done by someone, somewhere. When you see an effect you'd like to use on any website, anywhere, knowing HTML means that you can simply use your browser's 'View Source' function to see how it was done and adapt it for your own website - this is generally considered an OK thing to do. You'll also be able to start building up little libraries of code you've written yourself - a two-column layout, a splash page, and so on - and quickly adapt them as they're needed.

Even better, if you already have some HTML from a template that you bought or that came with some software, having built websites yourself will give you enough know-how to figure out how to edit it and make it look exactly the way you want.

Definitions of Note

Don't know your HTML from your HTTP? Your cache from your cookies? The web has serious amounts of jargon, and it seems like people come up with new words almost every day. Most of it isn't especially useful, but there are some words that it's good to know to help you along on the web. Here are the essentials.

Apache

The most popular web server. It is open source and free for anyone to use. Blog. A short for 'weblog'. A web page that is updated like a diary, with the most recent writing first. Usually done using blogging software instead of being maintained by hand. Browser. A web browser is the software that you use to view pages on the web. Internet Explorer is the most common browser.

Cache

A web browser's cache is where it keeps files that it has downloaded from the web and might need to use again. A site's logo and navigation graphics may be stored in the cache, for example, so that they don't have to be downloaded again each time you go from one page of the site to another. This happens automatically.

Cookies

Small files that websites can store on your computer to let them 'remember' you. When you log into a website and you're still logged in when you go back there later on, that's because the site gave your browser a cookie. Favorites. Also known as Bookmarks, this is a place in your browser where you can save links to pages that you'd like to visit again.

Flash

A browser plug-in developed by Macromedia that displays animations and animated websites.

FTP

File Transfer Protocol. The usual method of uploading files from your computer to a web server.

HTML

Hypertext Markup Language. The language that web pages are written in.

HTTP

Hypertext Transfer Protocol. Theoretically, the way that HTML pages are sent between a server and a browser, although in practice HTTP is used for sending all sorts of data, including graphics and file downloads. Many files should really be provided using FTP, but HTTP is considered to be easier and faster.

IIS

Internet Information Server. Microsoft's competitor to Apache, comes with versions of Windows that can be used as web servers. Often considered to be somewhat insecure and prone to crashing, although recent versions have improved.

ISP

Internet Service Provider. The company or institution that provides your computer with access to the Internet, usually in exchange for a monthly fee.

Link

A link is some text on one web page that will take you to another page if you click on it.

MySQL

MySQL is a free, open source database. It is often used for smaller web applications and websites.

Open source

Open source software is software which makes its source code freely available. This is intended to give you more freedom to modify the software however you want (or pay someone to modify it for you), instead of tying you to a company and relying on them for updates. In practice, this means that the software is available for download at no cost. Visit www.opensource.org for more information.

PDF (ebook)

Portable Document Format. A document format that aims to reproduce text exactly the way it would appear on a page. Viewable in web browsers using a plug-in, but disliked by many users because it can be very slow.

PHP

Stands for 'PHP: Hypertext Processor'. A very easy to learn and easy to use scripting language that is one of the most common on the web, helped along by the fact that it is also free. It is most often used in quite simple ways, such as retrieving text from a database and adding it to a page.

URL

Uniform Resource Locator. A technical term for a whole web address, such as <http://www.example.com/page.html>. It is called uniform because you can use similar addresses to refer to entirely different kinds of resources: for example, windows refers to your Windows folder, and ftp://ftp.example.com/public_html refers to a folder on an FTP server.

W3C

The World Wide Web Consortium (three Ws and a C, so W3C). This is the standards body that is considered to be 'in charge' of the web, and decides what gets put in and taken out of the various versions of HTML, amongst other things.