/ Lesson 1: Introduction to the World of Webpages

Lesson 1: Introduction to the World of Webpages

Welcome to Creating Webpages!

This course is designed to introduce aspiring Web developers to the wonderfully inventive world of 21st Century Web development.

If you enjoy being creative, the skills you'll learn in this course will be fun and profitable for many years to come. Of course, Web development isn't for the faint of heart. It can be a bit confusing and difficult in the beginning, but mastery will come over time.

Meet the Instructor

My name is Alan Simpson, and I'll be your instructor. My interest in the Internet began in the early 1990s when it first started entering public awareness. I created my first website back then. At the time I was a full-time author, and I wrote a couple of books on creating websites. However, the advantages of the Web and other forms of electronic publishing soon made me lose interest in print publishing and I've been emersed in Web development ever since.

We'll be covering a lot of ground in this course. Not all of it will be appropriate for computer beginners. So let's take a moment to discuss who can realistically benefit from this course.

Who is this Course For?

What you DON'T need to succeed in this course:

- You *don't* need any experience creating websites.
- You *don't* need any special programs like Dreamweaver—or even the knowledge that such products exist.

What you DO need to succeed in this course:

- You *do* need some basic computer literacy and skills.
- You *do* need to be familiar with terms like *desktop*, *folder*, *file*, *icon*, *click*, *open*, *close*, and *save*.
- You *do* need to be familiar with creating, editing, saving, and opening files.

If you're new to computers, or if you find it difficult to follow along in the first couple of lessons, you might consider live classroom training or a private tutor to get you through the absolute beginner basics before attempting this course.

But if you've been using computers for a while and you're familiar with the terms and concepts I threw at you in this section, then you should do just fine. Come on over to Chapter 2, and we'll get started!

The Web vs. The Internet

Before you start creating websites, you'll need to know what the *Web* is. This chapter will serve as a crash course in the basics. What most of us call the ***Web*** is officially called the ***World Wide Web***. The Web is a collection of billions of websites and webpages containing information. These websites and webpages can be accessed via the Internet.

Webpages and Websites

This course consists of multiple webpages—in fact, the page you're reading right now is a webpage. The online school through which you are accessing this course is a website consisting of many webpages. Other popular websites include Google, Facebook, Twitter, and YouTube.

The World Wide Web is a service of the ***Internet***. The Internet is tens of millions of networks across the globe. A ***network*** is a collection of computers that are all interconnected to one another through cables. However, networks are more than *just* computers and the cables connecting them.

The internet (and the networks within it) also consists of a set of ***protocols*** that define how different types of communication work. There are different protocols for different services. Here are some examples:

Text equivalent start.

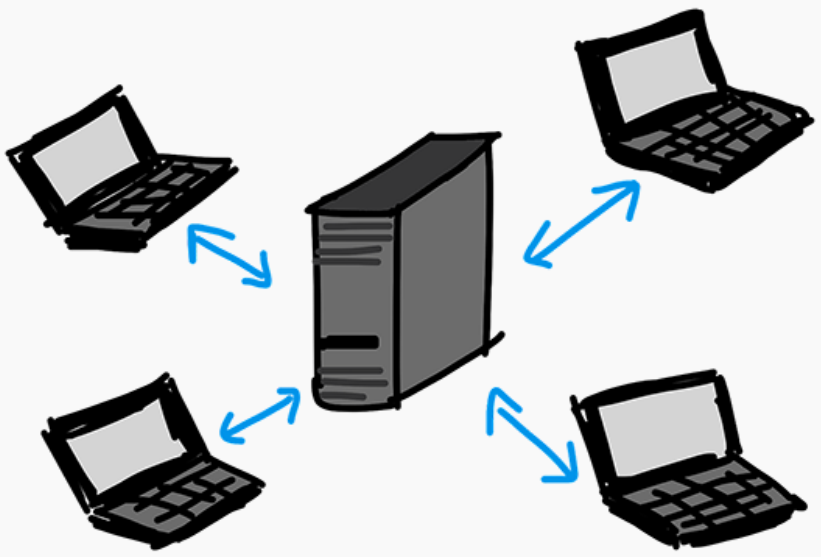
Topic	Information
IMAP, POP3, and SMTP	***IMAP (Internet Message Access Protocol)***, ***POP3 (Post Office Protocol version 3)***, and ***SMTP (Simple Mail Transfer Protocol)*** are protocols that define how email works.
HTTP or HTTPS	***HTTP (Hyper Text Transfer Protocol)*** and ***HTTPS (Hyper Text Transfer Protocol Secture)*** are protocol that defines how the Web works.

Read the topic in the first column. Then read the second column for the information.

Text equivalent stop.

How the Web Works

Slideshow Slide 1 starts here



The Web works through a system of ***servers*** and ***clients***. The servers are computers where the websites are stored. They're called servers because they "serve up" webpages to anyone who requests to see them. Those requests are made by clients. The client is the computer, program, or device that sends the request and receives and displays the requested page.

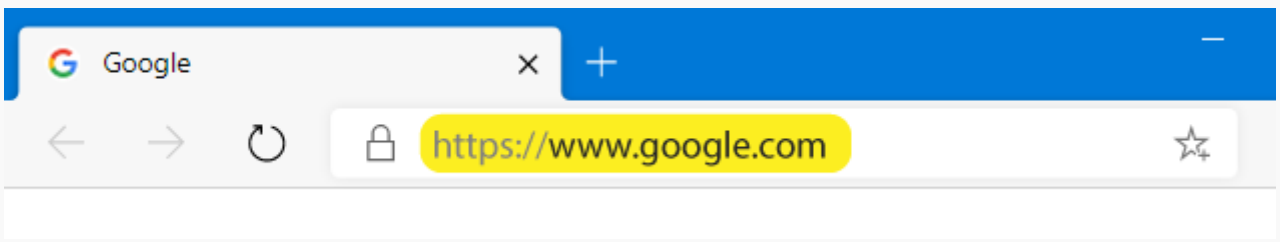
Slideshow Slide 1 ends here

Slideshow Slide 2 starts here

One of the most common clients is the ***Web browser***. Just as there are many brands of cars, toothpaste, shampoo, and everything else, there are many brands of Web browsers. Some common ones include Microsoft Edge, Internet Explorer, Safari, Firefox, Opera, and Chrome. But regardless of the brand name, they're all Web clients—meaning they all provide access to all pages on the Web via the Internet. In other words, you can get to any website in the world using any Web browser in the world.

Slideshow Slide 2 ends here

Slideshow Slide 3 starts here

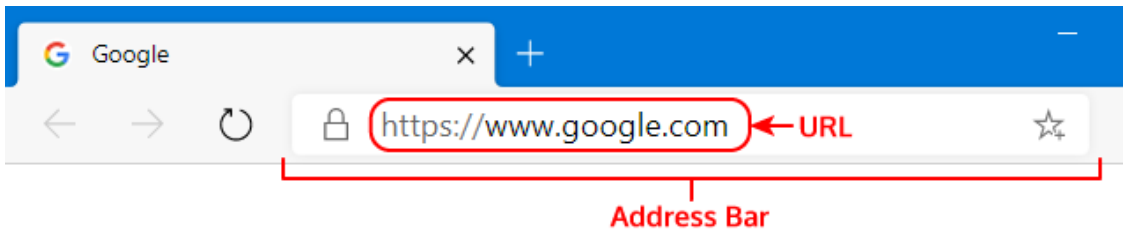


Every page on the Web has a unique address called a ***URL (Uniform Resource Locator)***. To view a page, a client needs to send that page's URL to the Internet.

Slideshow Slide 3 ends here

How URLs Work

So how do you send a page’s URL to the Internet? Sometimes all you need to do is click a link. If there's no link to click, you can type the URL into your Web browser's address bar and then press ENTER.



After you type the URL and press ENTER, the client (your Web browser) hands that URL off to the Internet as a request. In this case, the Internet gets the request to Google's Web server, which in turn responds by using the Internet to send the Google home page to your Web browser. Your Web browser receives the incoming page and displays it on your screen.

Even though every website has its own unique URL, they all follow a uniform pattern that generally looks something like this:

http://host.domain.tld

Let's break this down.

Text equivalent start.

Topic	Information
http://	The ***http://*** part is the extension for the Hypertext Transfer Protocol that we discussed earlier—the protocol that defines how the Web works.
host	The ***host*** is the prefix name of a server computer. Typically, the host is www (for World Wide Web), but the host name can actually be anything. This is also called a *subdomain* name.
domain	The ***domain*** part is the domain name. The domain is typically the organization that owns the site (such as *google*), although it can be any name.
tld	The *tld* part is the ***top level domain***, which is generally used to indicate the type of organization that owns the site. For instance, *.com* indicates a commercial site.

Read the topic in the first column. Then read the second column for the information.

Text equivalent stop.

When you type a URL into a Web browser's Address bar you start with the protocol extension *http://* with the host prefix *www.* followed by the domain for the page. The same basic idea applies to all browsers.

Note

Some browsers may hide the *https://www.* part once you're on the site, so in this case you'd see only **google.com** in the Address bar.

Common TLDs

TLD	Type	Example
.com	Commercial	http://www.microsoft.com
.gov	Government	http://www.fbi.gov
.edu	Education	http://www.harvard.edu
.org	Nonprofit organization	http://en.wikipedia.org
.net	Networking	http://www.html.net

TLD, Type, and Example

Web Pages

When you type the URL for a website without specifying a particular page for that site, you get the site's ***home page***. The home page is default landing page that opens automatically when you don't specify a page within the site. So, if you browse to apple.com (https://www.apple.com/) , you're taken to Apple's home page.

When you create a website, it'll have a primary URL that people can type into their browsers to visit. That URL will take people to your site's home page. Sometimes the site only needs one page, and that's fine. However, many times websites require more than one page. So, iff you wanted to browse directly to the iPhone page within the Apple website you would need to include the page in the url. This is done using a backslash followed by the page name. So, you would browse to apple.com/iphone (https://www.apple.com/iphone) to be taken to Apple's iphone page.

If a website needs more than one page, you can put ***links*** on your home page that visitors can click to access those other pages. These links can be part of the site menu, or they can be tied to specific text or pictures.

Creating a Website

Right now, you don't have anything to put on a website. To get started, you need to create one or more Webpages to put on the site. You'll create those pages on your own computer. Of course, you'll need a Web server and a URL to make the site accessible to the world. But you don't need that right now.

Later in this course, you'll learn you how you can get space on a Web server and a URL for your site. Then we'll give you step-by-step instructions for uploading your pages to the Web server for anyone to see. Best of all, it won't cost a cent!

Okay, let's head over to Chapter 3 and talk about creating a website.

What is Hypertext?

As mentioned earlier, the set of rules that define how the Web works is called the ***Hypertext Transfer Protocol***. You may be wondering what this bit of jargon actually means, so let's pick it apart.

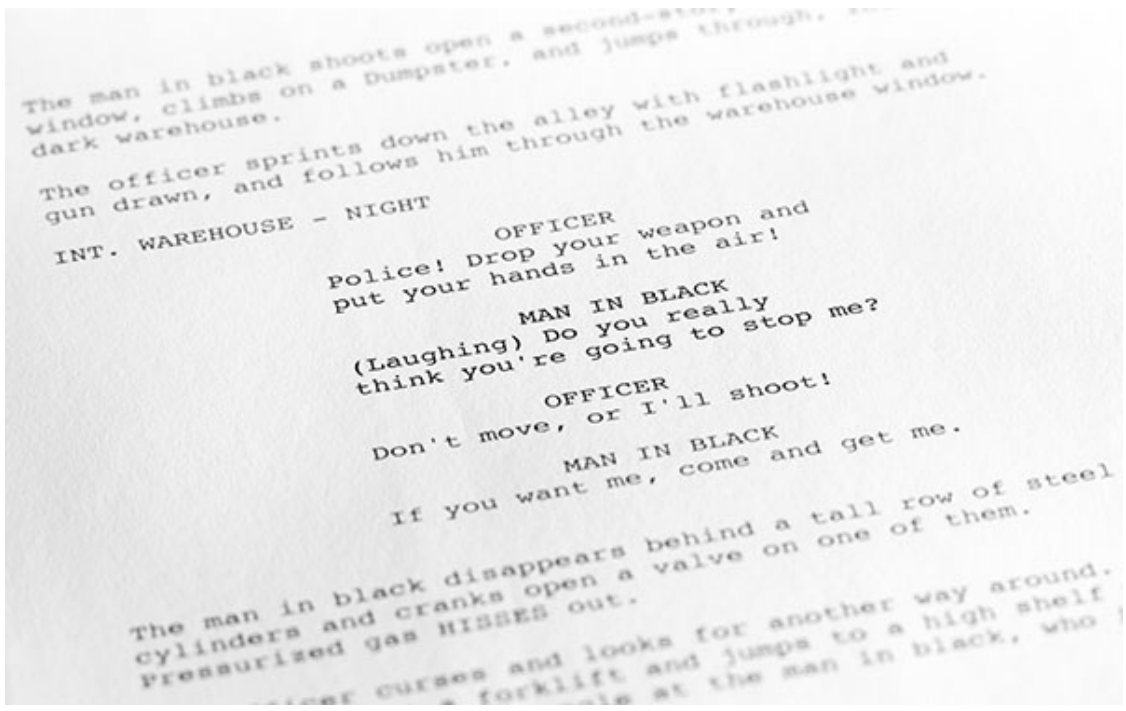
- **Protocol:** A protocol is a set of rules that defines how things are done.

- **Transfer:** When you request a Webpage from a Web server, that page has to be transferred (copied) across the Internet from one location (the server) to another (your computer). So part of the protocol defines how the transfer takes place.

- **Hypertext:** *Text* is words. The prefix *hyper* implies something above, beyond, or better than text.

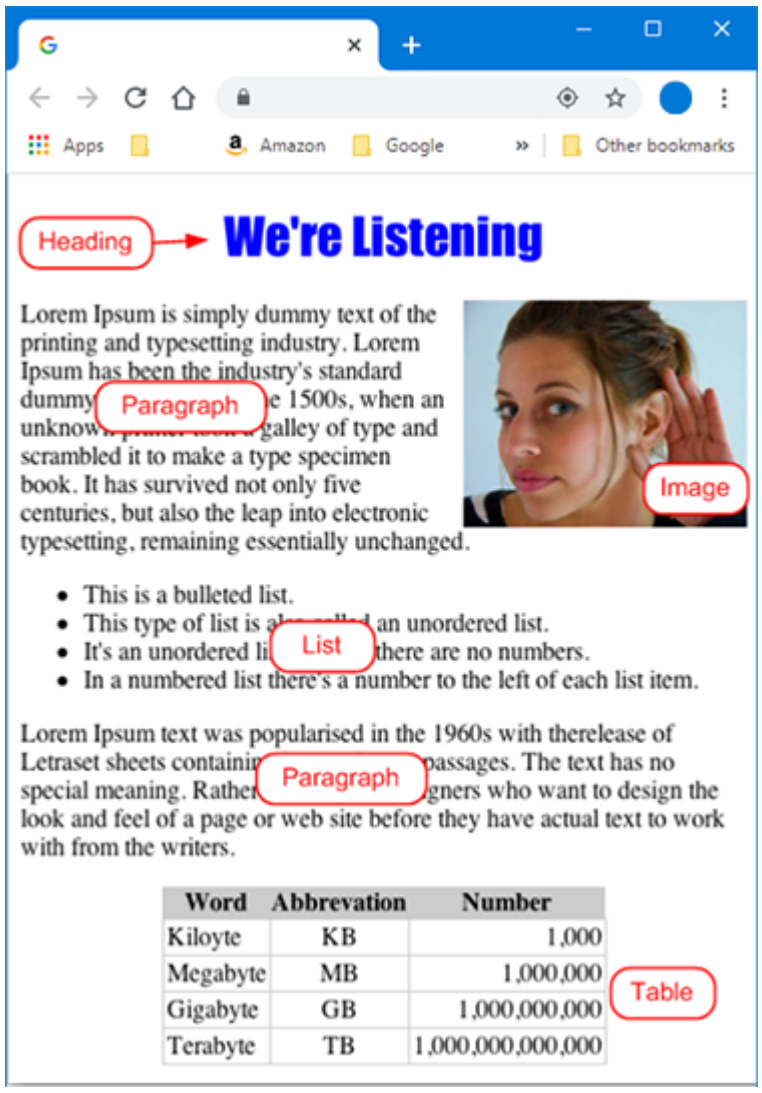
So what is *hypertext* exactly?

Well, let's say you grab a blank sheet of typing paper, stick it in an old-fashioned typewriter, and type a letter on it. What you end up with is a page of text. Plain, simple, old-fashioned text where every letter in every word is the same size and color (black), with no pictures or colors or anything else fancy.

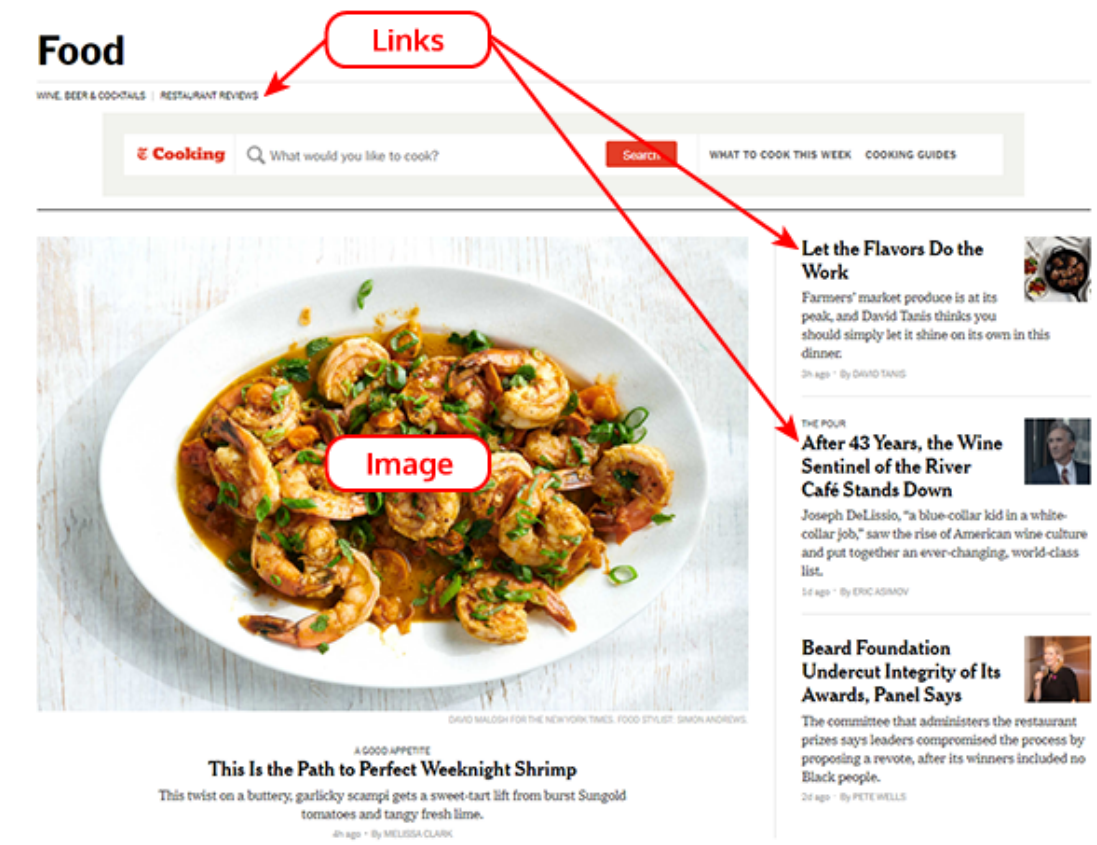


There's nothing "hyper" about the text in the example above. In fact, it's just about as plain as you can get. If you've spent any time at all on the internet, you know that webpages have a lot more going on than just plain old text.

Most webpages use the same ***design elements*** that you see in professionally published magazines and books. Some common design elements found in websites and print media include headings, pictures, paragraphs, columns, lists, and tables.



However, webpages also contain more dynamic design elements like links and interactives (such as slide shows). These make the presentation more interesting, and the information better organized. You can do a lot with pictures and links to provide an interesting and useful presentation.



Your pages can look pretty much however you'd like them to look. When you create your own pages, it's up to you to decide how you want to use design elements to make the best presentation possible.

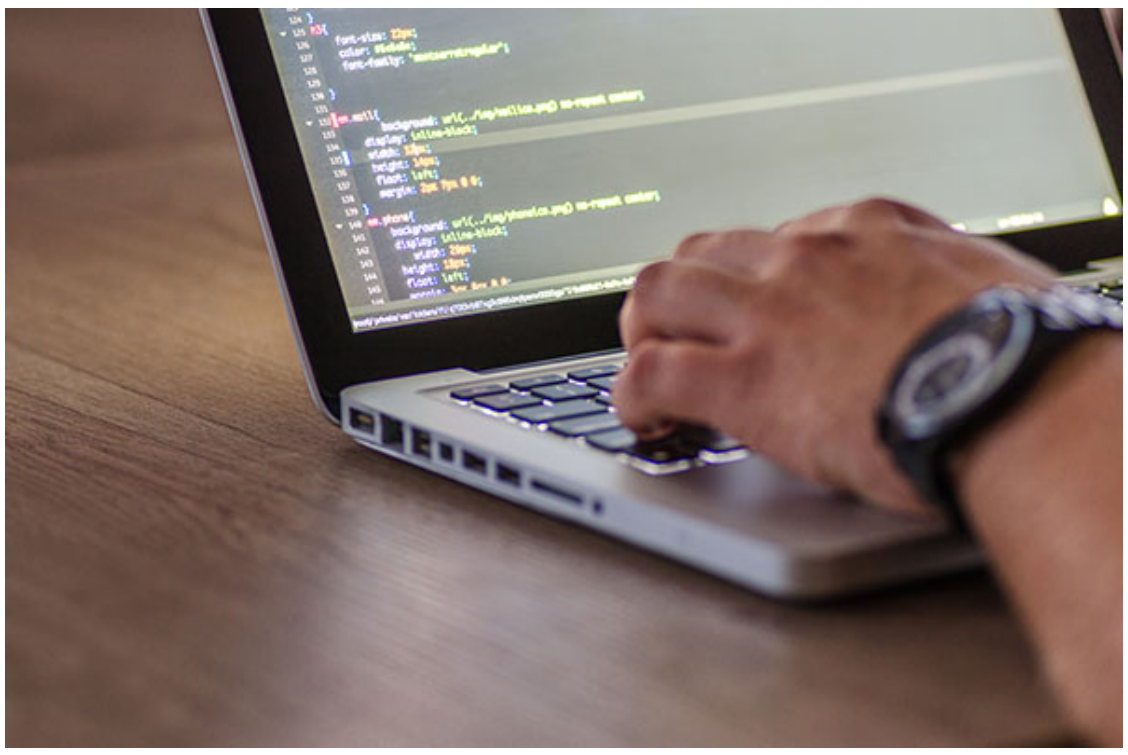
Elements

In the publishing business (paper or electronic), design elements are so commonplace that most people in the business refer to them simply as ***elements***. That's an important word that you'll be hearing for the rest of your Web design and development career.

The way you pick and choose your design elements is through a language called ***HTML***. All webpages (and in fact, many electronic documents including e-books) look and work the way they do because of HTML.

Introducing HTML

HTML is abbreviation for ***Hypertext Markup Language***. (There's that word *hypertext* again!) As its name implies, HTML is a language that allows you to *mark up* your text so it's no longer plain boring text like you get from a typewriter. Rather, your creation is text embellished with elements like pictures (images), links, headings, lists, paragraphs, tables, and many other kinds of things. Most of this course will be about HTML (and its twin XHTML). So let's take a moment to discuss what that's all about.



HTML (as well as anything else that concerns how the Web works) is defined by an organization called the World Wide Web Consortium (https://www.w3.org/). People in-the-know often refer to it as the *W3C* or even just *W3*. HTML is defined through committees at the W3C. Committee members come up with ideas and put them out on the Internet as suggestions. The people who create Web browsers and websites discuss it.

HTML is more of an evolving standard.

As you know, getting groups of people to agree on anything is generally a long and slow process. With computer languages like HTML, it's also a process that never really ends. There isn't one HTML language in the world. HTML is constantly being expanded and improved so that we can all build better websites running on, faster, smaller, cheaper, better computers and mobile devices (cell phones, tablets, e-readers).

Over the years, the committees have released quite a few versions of HTML:

Version	Year
HTML 1	1991
HTML 2	1995
HTML 3	1996
HTML 4	1997
XHTML 1	2000
XHTML 1.1	2001
HTML5	2014

HTML Versions and Years

As you can see, the name changed to XHTML around the turn of the 21st century, and now its back to HTML with HTML5. You don't need to be too concerned about the reasons for all these different versions and the differences between them right now. The similarities between the versions far outweigh the differences.

Now that you have a bit of background and general knowledge, let's move on and start gearing up to create your first website. In this course, you'll learn the core HTML that applies to virtually all versions, with special emphasis on HTML5, the current standard.

Getting Ready to Build Your Site

Every website that you can visit on the internet exists in a ***folder*** (also called a ***directory***) on some Web server. A folder is a place where you can store files.

Most of the time you'll start by creating your site on your own computer. This way you spend some time developing and perfecting it, before launching it out in the view of the public. Once everything is looking good, and you are ready to *publish* the site, you copy the files to the Web server's folder. At this point it becomes a ***live site*** on the Web.

Creating a Folder for Your Site

We're going take that very first step of creating a folder for your site. The folder on your own computer that contains all of the files that make up your website is your ***local folder***. We'll provide step-by-step instructions for Windows and Mac users—just in case you aren't quite so familiar with the terminology or steps.

Local Folder

Web professionals refer to it this way to distinguish it from the ***live site***—the folder on the Web server where the published copies of the files reside.

Your local folder can be anywhere on your computer, and you can name it anything you like. However, you want to make sure it's easy to get to and that it's named something logical for the project you're working on. You'll be using it often as you create your site.

For this course, we'll name the folder ***MyWebsite***. You can save it on your desktop or put it in your **Documents** or **My Documents** folder. In the next sections, we'll go through the steps of creating a folder using Windows and a Mac.

- **If you want to put your folder on your computer's desktop,** make sure that you can see the desktop.
- **If you want to put the folder inside in your Documents folder (or another folder),** open that folder.

Note

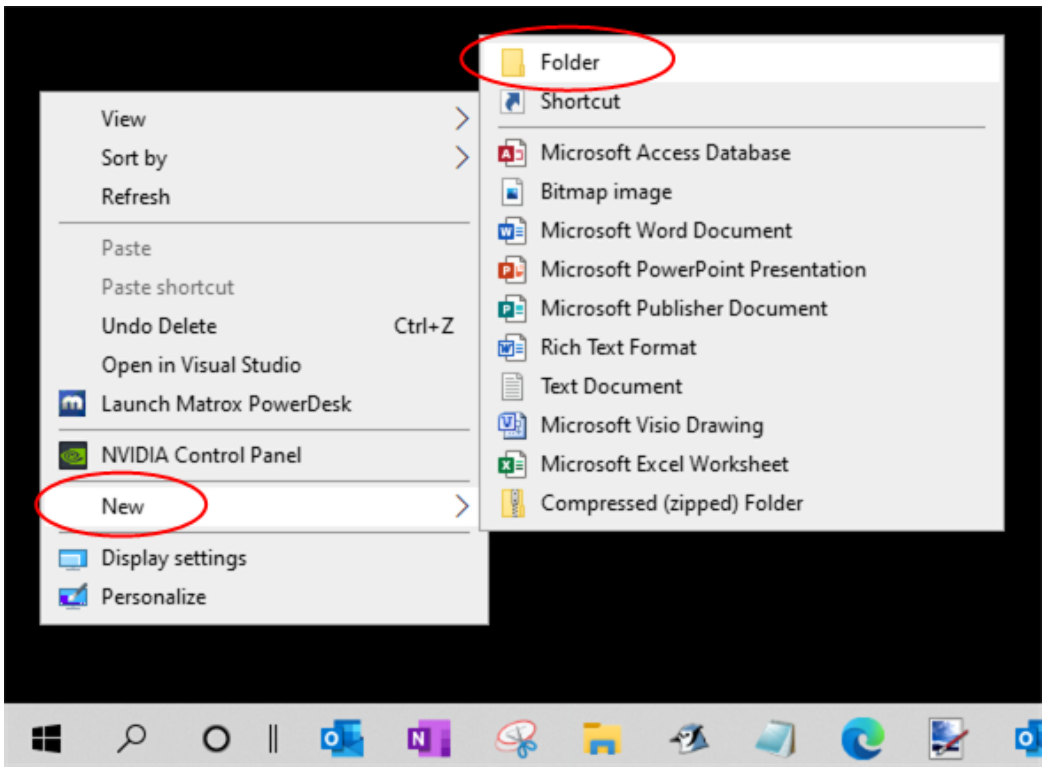
What you see on your computer, will vary depending on the whether you're using a Windows or Mac, the version of the OS you're using and your installed programs. While we can't give steps-by-step instructions for every version of every operating system (OS) ever created, there should be enough here for you to follow along.

Important

If the instructions are too advanced for you, it's possible that this course is going to be too advanced for your current level of computer literacy and basic skills. In that case, you might want to consider trying to find resources appropriate for your current skill level and the computer and operating system you're using.

Here are the Steps for Windows

1. Go to your Desktop (or your Documents folder).
2. Right-click an empty spot on the desktop or in the current folder.
3. Choose **New > Folder** from the menu that appears.



4. Type *MyWebsite* for the new folder name.

5. Press **ENTER**.

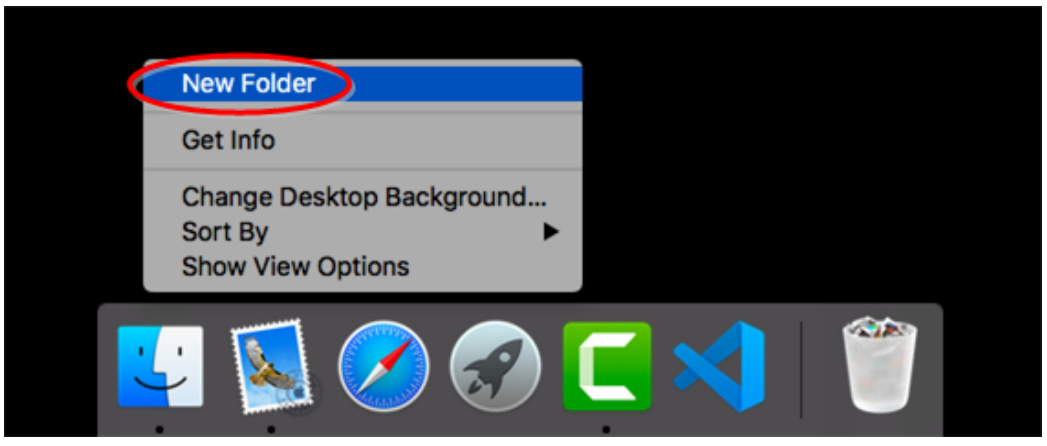
6. When you're done, you should see the folder icon with the name you provided.

Here are the Steps for Mac

1. Go to your Desktop (or your Documents folder).

2. Press **CTRL** + click an empty spot on the desktop or in the current folder pane.

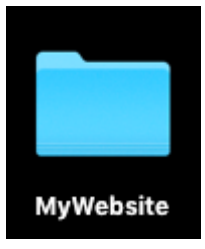
3. Choose **New Folder** from the menu that appears.



4. Type *MyWebsite* for the new folder name.

5. Press **ENTER**.

6. When you're done, you should see the folder icon with the name you provided.



Both Windows and Mac display icons for folders as little manila file folders. That's because like paper manila file folders, a computer folder is a place where you can store files. The exact appearance of the folder icons depends on many factors, including what version of Windows or Mac OS you're using, the size of icons at the moment, and the background color behind them.

Hopefully, creating the folder was easy for you. Next, we need to talk about actually creating your pages, which requires a text editor or other authoring program.

Choosing an Editor

As we discussed, every website consists of one or more Webpages. Every Webpage is a plain text file. That basically means that when you open it in a text editor (or Web browser), you see text characters such as the letters, numbers, and punctuation symbols that you can type on a keyboard.

A nontext file can contain a lot more than that because it contains information for programs, not people, to process. Opening a nontext file in a text editor or Web browser can create a jumble of characters and symbols that appear to have no rhyme or reason behind them.

```
/ ▯½)(<{À▯£
œq´B…▯f¤¢}¢ÑÒ·‡VUfž▯´Gšâ¸D▯▯¼
6( ù‰¡ zà©ðœBÑ›,Â3À+▯*ô¯PÁÙ@♠&1!è▯^3ÊZTçñºÒÄmùà<▯Œçïw▯J▯ŠP▯▯JÀŸþô§▯5à_Ç▯ýË¿¼ÿ—,
?▯ºaÕ*÷?▯hÞÜdÝª]ûZß¾G§LÙôÝw¢ùÂ`ÿ—_z▯▯Ò¸qÎ¿þ«I“Ï}øá–¯¿6ê)(<;0ã¾þúkxÎ8▯ŸŠP▯▯J ▯1ã

▯¡èMAá▯c»Ø’♠ûèa%¼å?LO¿=>Í÷ƒ4ß‘©||F¦xHñþ0åæ‡É7‡'{
Oº1,õöW▯†+O ŒƒÛèÛFT♠§`
▯½)(<K|bà¶¬¢vÌ²–Þr“Ó|‡åE~›ûC~Ìªüè•ùÑËó£¿ÏZ–▯õ-ŸÜPÇ¤ëÃòrS▯Ê▯`µªU▯▯nèP£DA¡x(zSF
â7▯Äo,ˆßP▯·© îÇ,Ø▯µ▯q?äE~tý▯Ëô———””▯▯▯▯▯žžnÌx▯>LMMÍÍÍ5ž▯P¤º¬€▯î¥—
▯îÆ
£DA¡▯(zSPxfÐí†ñÌ▯ÖÓ[*ô▯³ªÛâÖ▯▯▯T▯±Å¬.@▯»*/|Ñý’è-44´yóæ¯¿þz▯▯5êÖ-»nÝ:HË˜§C``à;ï¼s
Ë▯¦NU▯œ,5Pô¦ ðÌ ▯u+2t▯O@o>#ó£W▯Æm&Ü▯³,O^¸óýk%Ð›¿¿▯«V-öïßO ▯222Œ▯▯▯µ6mÚœ9sÆxn ôf.

Ï♠òëØŒg¦º’Pòr“S¼GäG//|&iÂmÑß▯Ä,Ï▯wJ¼:,ï%z»wï^§N®_¿n<▯„øøø¨¨¨ØØØ¼¼<NQëØ±ã¹sç´
ôö-ñÌ▯O@o7?ÌZVø.‰        ·E▯[▯½,/Ì1ñÊÐ¾Ü4c¢▯▯▯'U¬X▯2kÐ ÁÉ“'!¼6mÚ1Ü¸1&&æÈ‘#íÛ·÷
QQQ▯[Ÿ>},ƒƒ▯▯    é +y[rãÆaaaÎÎÎUªT¹xñ"\H▯'''â?ÜNOO'°û#_ªœ3§Û^z©àèQ£DA¡x(zSPx
`8Ô<==ÉÝ¹sgõêÕkx®=cÆŒ=z\¼zUŠ4lØ▯yÍš5çÍ›wûöíþýûëÍþN V▯:´Ûø1Üh▯*(X„¢7▯…g:~
'Éð{o,'£·ˆÅ▯1Ëã¶ÈÅ▯Q.y¡ó▯.öÏ
œpÙÒË“ØS‘;0Õcbb8Ê©^t||<▯▯QZZZšV▯‡) pûáÃ‡ú¬ß       Äj²▯▯qÛöíF¡,BIPô¦ P*xùÑ·¼ù@u
œ▯w¼wÎÝOã.½÷"Ó›žØ^~¹ÀÙÙ(WP°▯ŠÞ▯▯J▯Ìª©:’{X³f–³sáOQ▯ƒüÜ´Ä▯£▯¯¼—xõƒÄ+C▯<▯?ñ—‡Ä_▯▯wyh
```

You don't want your Webpages looking anything like that mess. Because anyone who visits the site and sees something like that on their screen isn't likely to come back in the future.

There are hundreds of text editors and other Webpage authoring environments ranging in price from free to many thousands of dollars. For this course, we’re going to use free and simple text editors that everyone has. This way, you won't need to buy, download, or install anything. Exactly which you'll use depends on whether you'll be using Windows or Mac to create your website.

Note

This course doesn't cover Linux, UNIX, or other platforms. You can certainly use such computers to create websites, but there isn't enough space in this course to offer step-by-step instructions on the absolute basics like creating folders and using a text editor for those operating systems.

Text Editor for Windows

If you're using Windows, *Notepad* is the built-in text editor that comes with every version of Windows. You can start Notepad the same as you would start any other program in your version of Windows.

Notepad isn't the same as WordPad.

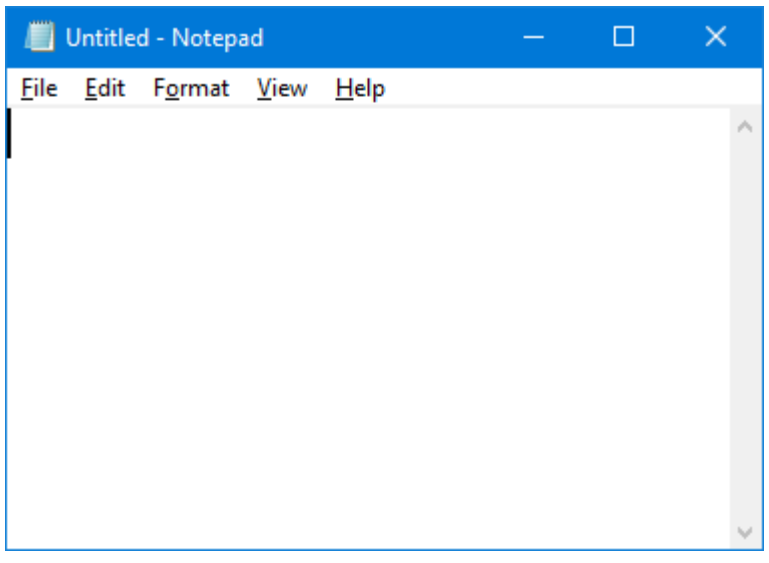
Notepad is a text editor and WordPad is a word processing program (like Microsoft Word). Be careful not to confuse the two. Word processing programs aren't appropriate for creating websites. If you use Word or WordPad, your page might end up looking like the mess shown in the figure above.

Here are the Steps

1. Click the **Start** button in the lower left corner of the screen.

2. Start typing the word *notepad* to search available programs.

3. Click the **Notepad** icon to open the app.



Pin the App

If you right-click the **Notepad** icon, you can choose **Pin to Start** and **Pin to Taskbar** to make it easier to find the app in the future.

You don't have to do anything in the program right now. Just make sure you can start it and recognize it when you see it. For now, you can close it as you would any other program, by clicking the **Close** (X) button in the upper right corner, or by choosing **File > Exit** from its menu bar.

Text Editor for Mac

If you're using a Mac, you can use *TextEdit*, which comes with all Mac computers. However, there are some settings you'll want to change to use it as an editor for creating Webpages.

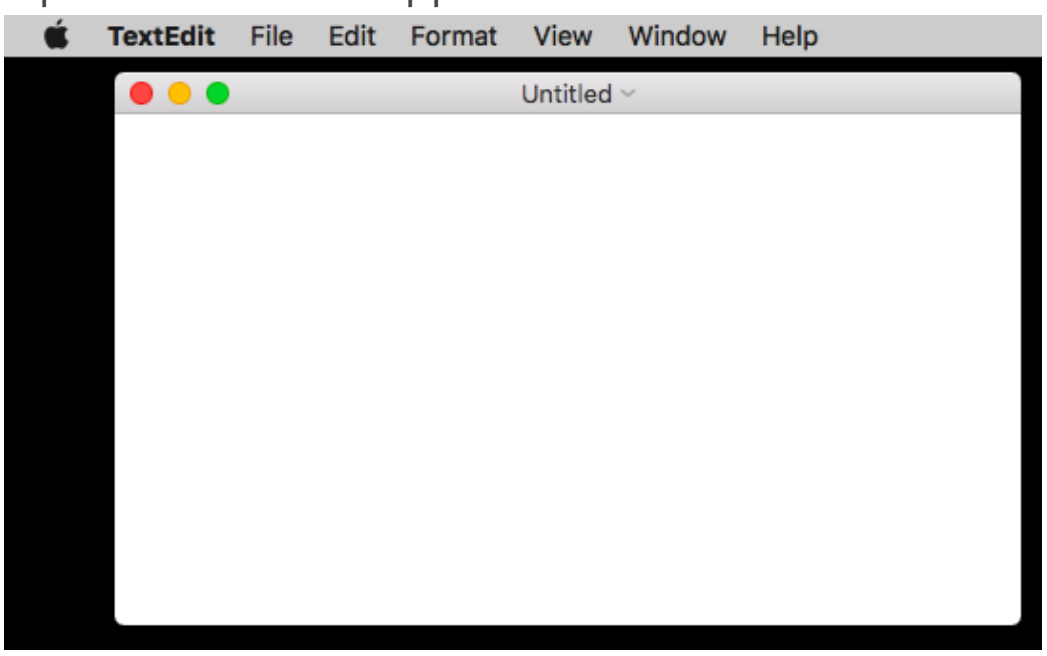
Important

Don’t skip these steps if you're using a Mac! If these steps aren’t performed correctly, you may find the hands-on activities confusing and difficult. Make these adjustments right now before you move forward.

Here are the Steps

1. Go to the **Applications** folder or **Dock**.

2. Open the **TextEdit** app.

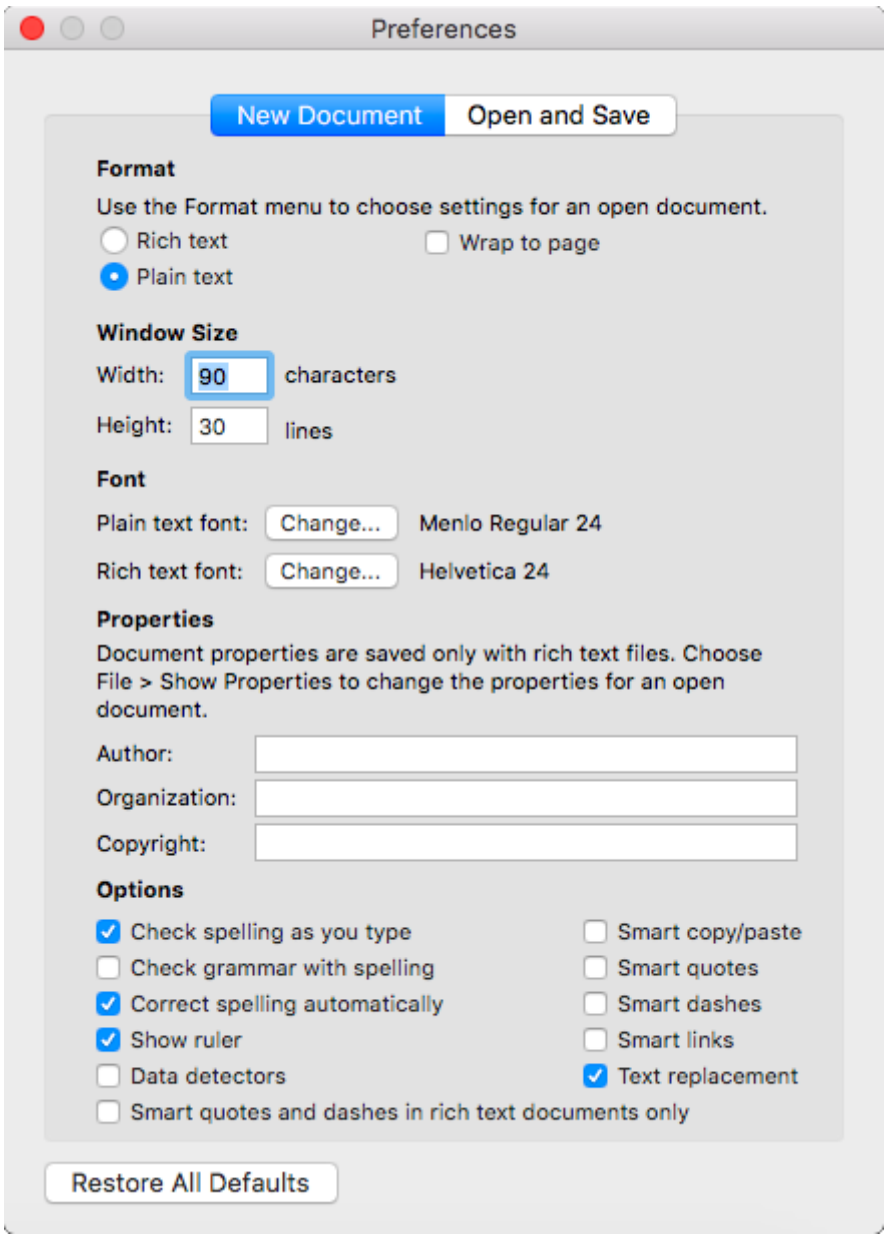


Note

If you're using a newer cloud-enabled version of TextEdit and see a **Documents** dialog box, click **On My Mac** in the dialog box that opens. Then click **New Document** to open the editor.

3. Once you've opened TextEdit, click **TextEdit** on the Application menu and choose **Preferences**.

4. Go to the **New Document** preferences.



5. Under **Format**, select **Plain Text.**

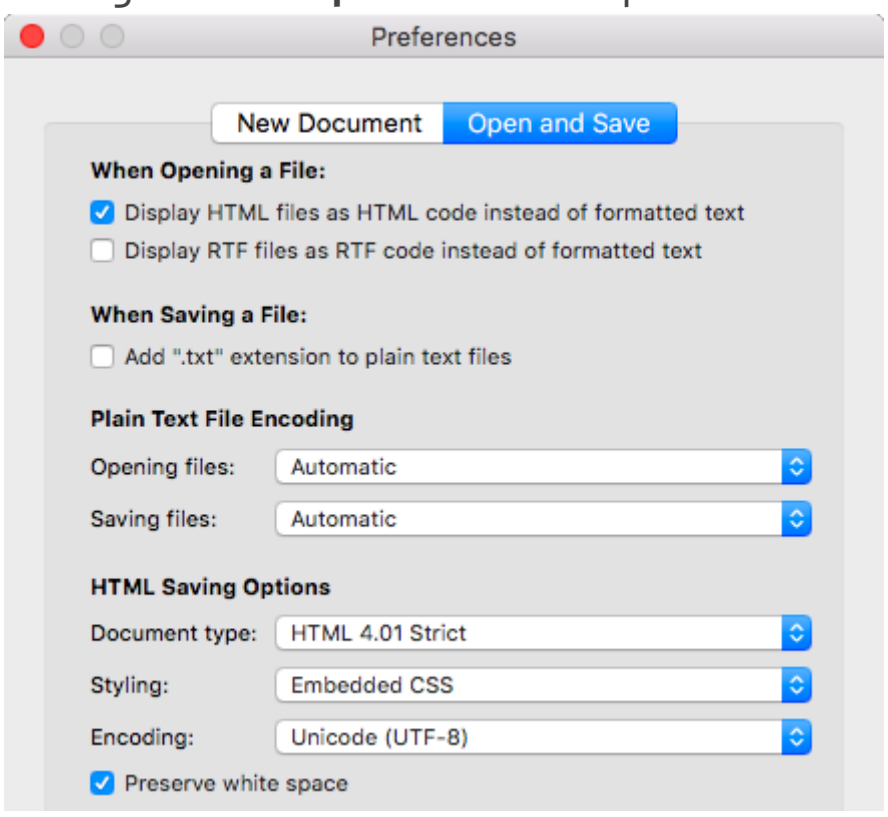
6. Under **Options**:

   - Select **Check spelling as you type**
   - Select **Correct spelling automatically**
   - Select **Show ruler**
   - Select **Text replacement**

Note

Not all Mac OS versions are the same, there may be other options you'd like to adjust in your **New Document Preferences** dialog box.

7. Now go to the **Open and Save** preferences.



8. Under **When Opening a File**, select **Display HTML files as HTML code instead of formatted text.**

Note

Depending on your version, it may be worded as **Ignore rich text commands in HTML files** or something along those lines.

9. Under **Plain Text File Encoding**:
    - Set **Opening Files** to **Automatic**.
    - Set **Saving Files** to **Automatic**.
10. Under HTML Saving Options:
    - Set **Document Type** to **HTML 4.01 Strict**.
    - Set **Styling** to **Embedded CSS**.
    - Set **Encoding** to **Unicode (UTF-8)**.

Note

Again, not all Mac OS versions are the same, there may be other options you'd like to adjust in your **Open and Save Preferences** dialog box.

11. You won't need to use TextEdit until later, so go ahead and close the **Preferences** window.

Let's wrap things up with a quick review and talk about some general features of this course!

Lesson 1 Review

In this lesson, you learned some important terminology and concepts related to creating websites. Even though, to some extent, we all hate learning terminology and concepts, there's really no getting around it. You can't live and work in a vacuum. Eventually, you're going to have to mingle with, or learn from, other developers in the field. And doing that successfully is going to require that you can "talk the talk" and understand the language.

You did get a little hands-on practice when you created a folder for your site. Not exactly Web development, per se, but a necessary step. You'll do a little more hands-on prep work in the assignment for this lesson. In Lesson 2, we'll walk through creating a webpage.

Learning Check

Words to Know

Loading flashcards...

To finish the lesson, you'll need to complete the steps outlined below. Simply click "Next Up" at the bottom of the page to access the next activity. Or, if you wish to skip around, click the Book Icon in the top-right corner. There you'll find links to all the activities in this lesson. Here are your remaining activities:

- Check out the FAQs. Since learning something new usually raises questions, every lesson in this course comes with an FAQs section.

- Do the Assignment. Get some hands-on practice applying what you've just learned.

- Participate in the Discussion Area. Ask questions about anything that came up in the lesson, and share your insights. This is where we'll create a learning community.

- Take the Quiz. Reinforce what you learned with a short assessment.

- Browse the Resources for Further Learning section. Here you'll find links to helpful online resources relating to the lesson.

- Be sure to look at the additional resources. When it comes to a topic like this, there's always more to learn. You can find the link to these resources by clicking on the Resources link on the Learning Path.