A Brief Introduction to Web Development

By John Beckett
Professor of Computing
Southern Adventist University

Support Web Site: http://computing.southern.edu/jbeckett/WebDev1

"The heart of the wise leads him to acquire more knowledge and his ear seeks knowledge through listening."
Proverbs 18:15 (*The Clear Word Bible,* a paraphrase translation by Jack Blanco)

Don't print this book (all at once, that is)! Print each chapter as you need it, because the book is often updated as students ask questions. Better yet, use a second screen instead. Don't have one? Talk to me!

Version 9.00 September 11, 2023

Acknowledgments

Many textbooks acknowledge students as their primary inspiration, and this is no exception. This book grew out of their frustrations. I am also grateful to my wife for her endless support, and the best colleagues in the world - the faculty of the School of Computing at Southern Adventist University. The logo and icons were ably designed by Rebecca Johnson. Most of all, however, I am grateful to God who is the ultimate source of all knowledge.

Apology

So, why am I cluttering the planet with yet another introductory textbook for HTML? There do seem to be plenty of them around.

To understand this, you need to know the intended audience for this book. It serves two audiences, which to some extent are just two ways of looking at the same group of people:

1. Students who need an hour of academic credit in a computer skill.
2. Students who want to get going on the Web as a tool for communicating about some ministry or business they wish to develop.

So that's the "who" – but what about the "why" of another book?

Because we need less. We need people who want to get into Web Development to know less HTML because intensive focus on HTML among Web developers makes us dependent on them. The Web is cluttered with sites that were created, then left behind – a legacy of corpses that document good intentions.

Because we need more. We need people learning Web Development to use Content Management Systems so that they will leave behind Websites that can be maintained by custodians of the information. It's time that people updated their Websites instead of mumbling that they need to call in the Web person to update it. But try as they might, CMS writers have never totally replaced basic HTML skills. So we still have some HTML in this book.

Because we need to fit into the academic cycle. This book is structured as fourteen lessons, which fits into a semester cycle. It is used for a one-semester-hour online class.

Because we need wider counsel. I'm not the best at everything (not even in my own mind). Beginning with chapter 7 I refer the reader to excellent tutorials on the Web. In the process, students learn to use the Web as a resource for creating the Web. And they learn how to integrate information from multiple sources.

Finally, some will abhor the religious content in my examples. No "Lorem ipsum" here – I use real words, sometimes about important things. Understanding religion is essential to understanding history and the people around you. I've selected the text examples to be useful to a wide spectrum of readers (possibly excluding extreme fundamentalists at both ends of the spectrum). If you happen to be an Atheist, please be understanding. People like me are capable of pronouncing terms like "Pre-Cambrian" without losing our moorings. You'll survive, too. Who knows – you might even see the sense of where I'm coming from.

Preface

The purpose of this section is to help you decide if this book and the class it supports, are right for you. If you are reading this, you're probably thinking of building a small Web site. That's what we're here for! Using this book, you will learn:

- The basic mechanics of creating a Web site and making it available to people
- How to organize materials so that people can navigate your site
- Key aspects of HTML and CSS you'll need
- How to work with forms used to obtain information from users, and how to pass this information to a back-end program
- How to use your skills to exploit the capabilities of Content Management Software (CMS) systems you'll most likely be using

Teachers: This book is the textbook for a class I teach – CPTE 110 Intro to Web Development, a 1-credit course at Southern Adventist University. Boxes like this contain information unique to our environment. If you teach a course in which you would like to use this text, you will need to change this information. Remember to obtain a Creative Commons license to show it is a derived work. Localization information is a major reason I chose to write my own text. At this level, localization issues are a major barrier to student learning.

To the student: You may be tempted to just work through this book and learn it on your own. That's fine. But if you actually pay tuition, you get more. Here are some reasons you might consider enrolling in my course by calling 1-800-SOUTHERn:

- You get interaction with a teacher who helps with problems and holds you accountable for deadlines (I'm better at the former than the latter).
- All the infrastructure needed for this stage of your learning is ready-to-go including my Homework Server. Bring your highspeed-connected PC or Mac, and you're good to go.
- You get credit from an accredited academic institution – credit that is transferable elsewhere.
- As of this writing, taking this course qualifies you for free downloads of a wide variety of Microsoft™ software for personal use through the Microsoft MSDN Academic Alliance.

Beware of "Optionitis Interneticus" – my term for confusion that comes from so many ways of doing virtually anything to do with the Internet. This is very confusing to a beginner. This book focuses on <u>one</u> way to do things. I can guarantee that in most cases there are alternative methods that would work. Please understand if I don't cover them all.

What Kind of Computer Do I Need? Generally, a computer with a gigabyte of RAM memory and ten gigabytes of free space on the hard drive will do just fine.

The **computer lab** in Hickman Science Center room 1303 is an excellent place to do projects for this class. The computers are set up with the right software, and the lab operators may be able to help you with your coding. Bring a USB drive (and don't forget to take it when you leave!)

What Kind of Internet Connectivity Do I Need? An analog dial-up connection really isn't going to be good enough unless you are a *very* patient person. Any DSL or cable modem or dormitory connection will do just fine. Satellite or cell modem connections will be slow, but will work if you are patient.

Finding Things In This Book

What's with all the boxes?

The boxes in this book have various specific meanings.

The school logo indicates that this box is very specific to our class environment at this institution. Someone adapting this book to use elsewhere should replace the logo and change the content of these boxes. If you're using this book somewhere other than Southern Adventist University and it shows the Southern logo, it needs to be "localized" by your instructor or T.A.

This icon indicates that this box contains important information – usually something that has cost students a great deal of wasted time in the past, if they ignored or didn't understand it.

This is general information that explains something that is important for understanding it the topic at hand.

This is a note for people who use a USB drive for data storage

This box has information you might need if you are using Mac OS or Linux.

Here are specs for a homework assignment.

Metaphors

One of the best ways of teaching computers is metaphors. These are italicized so that you can skip them if you already understand the concept.

Source Code

When showing HTML code or other things you might type into your editor, this book uses Lucida Console font. Like this:

“`<title>This title will appear at the top of my browser window</title>`”

HTML and XHTML

For the purposes of this class, the terms HTML and XHTML are equivalent. XHTML is a version of HTML that follows stricter rules, and it is really XHTML we teach.

Table of Contents

Lesson 1: Getting Started

In this first lesson, we are connecting several things together. The goal is to create a very simple Web page and place it on the World Wide Web where it could be accessed by anyone (who knows the right username and password.)

What We'll Learn

- Where to get all the software you'll need – FREE!
- An important configuration change you need to make on your computer
- How to organize your projects for this class so you don't have problems later on
- The importance of proper filename capitalization
- A brief look at the architecture of the Internet
- Directories
- A sample minimal HTML document
- How to create a very simple Web page
- How to upload a Web page to the server
- How to verify that your page is correct

Don't try to use Microsoft Word to create your Web pages. It wasn't made for that – even if they do have a half-hearted attempt at HTML export in the SaveAs.. menu. Each year somebody learns that wrestling HTML out of Word is a lot harder than learning Visual Studio Code for this class. Trust me: you don't have enough time. Don't use WordPad either. It is possible to use Notepad, but Visual Studio Code is much easier.

SOUTHERN
ADVENTIST UNIVERSITY
School of Computing

Links to obtain these software items are located at:

http://computing.southern.edu/jbeckett/WebDev1/.

Loading Software

Everything you need is freely downloadable..

Begin by installing these tools on your computer. In some cases you may be asked if this is a legitimate program to install. It's OK.

An Important Configuration Change

Microsoft™ has selected thousands (maybe millions) of default behaviors for Windows. They got most of them right. One of them, however, just won't work for this class. But you can fix it like this:

1. Bring up a File Explorer window
2. At the menu along the top, click "View"
3. Click "File Name Extensions"

Mac OS Macintosh has something called "Text Edit." Do not use this program, as it will not work properly with HTML. Use JEdit instead.

Mac OS has something called a "folder" – which for our purposes is exactly the same thing as a directory. Mac OS

Organizing your projects

Here's how I do it. Even if you think you know a better way, consider doing it this way because this entire book assumes you've done it as follows:

Create a directory named "Web Homework" and build directories under it for each of the 14 assignments in the class. Don't know how? Here's how to do it in Windows.

1. Click the "Start" button in the lower left-hand area of the screen. Select "Documents" or "My Documents" depending on which shows.
2. Make sure the window that pops up is wide enough to have some blank space on the right. Now right-click in that blank space, select "New", and click "Folder." Sometimes it takes me a couple tries to get this right the way the windows pop up.
3. The new folder name will appear, and it will be selected. Type "Web Homework" in that name. Press the Return/Enter key. Now you have a place to put your work.
4. Double-click on the Web Homework directory you just made. That will navigate you into the folder itself. Inside it, use the same procedure you just used to make fourteen directories – one for each assignment in this class. Name them HW01 through HW14.

You must be absolutely consistent in your use of upper and lower case letters in file names when doing Web stuff! Get used to the idea that "a" and "A" are not the same letter when they're used in file names. I know, Windows lets you get away with inconsistencies (usually) – but Web servers are rarely forgiving. They are usually based on some form of Unix or pretend they're Unix in this regard (which is what IIS does), and Unix thinks a small letter and its capital are totally different things. Even if you use a Web server that lets you get away with capitalization inconsistencies, if you're sloppy about file name capitalization you're developing bad habits and your site can't easily be moved to another server.
Huh? I'd better give you some examples.

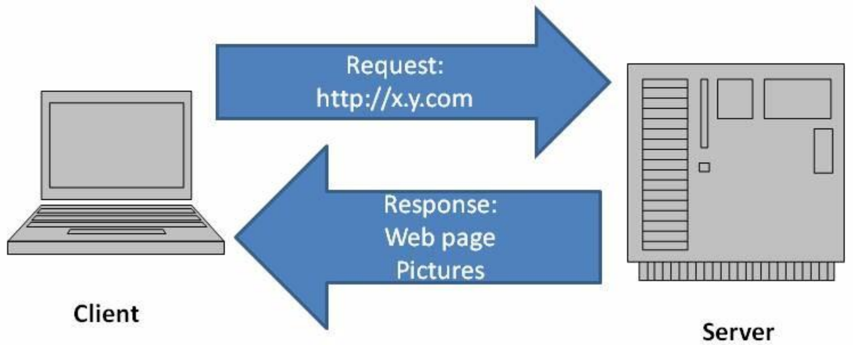
Actual File Name	File Name in HTML Code	Situation
`Abracadabra.html`	`Abracadabra.html`	Fine – they are spelled exactly the same
`Abracadabra.html`	`AbraCadabra.html`	Windows will think they're the same, but a Web server won't because the "C" is not the same as "c." Yes, it's that picky!
`Abracadabra.htm`	`Abracadabra.html`	Not the same. Notice that the slight difference in the file extension. You can call it either, but you must be consistent.
`Logo.jpg`	`Logo.JPG`	This will look fine on your Windows machine, but when placed on the Web it won't work because the extension is not the same. Again you can call it either, but you must be consistent.

The directories you've just made will serve as containers for the assignments in this class.

Spaces can be deadly in file names, especially if they are at the beginning or end of the name. Never create a file with a space at the beginning or end, or you may have serious problems managing files on the server.

Architecture of the Web

The World Wide Web is so complex that no one person can know how it all works. So this is a very simplified explanation. The Web is focused on several entities:



1. **Client** – the personal computer you are using.
2. **Browser** – a program you run on your computer. The most popular browser is Microsoft™ Internet Explorer. Other popular browsers are Mozilla™ FireFox and Apple™ Safari. This book uses FireFox for its browser, although we mention others from time to time.
3. **Internet** – a vast network of connections and equipment that connects your client to other locations anywhere in the world.
4. **Server**. This is the where Web pages are stored. Most likely, it runs either IIS from Microsoft or Apache, an open-source Web server. Our Homework Server uses Apache.
5. **Domain Name System** (DNS). This is a world-wide database that is used by your client to translate names of servers into numeric "IP" addresses (*a little like latitude and longitude*) for finding them physically on the Internet. DNS usually functions invisibly.
6. **URL** – A Universal Resource Locator is the term for the formal name for a Web site. It usually starts with http:// or https://.

People often use the terms "World Wide Web" and "Internet" interchangeably. That is a technical error, although there is usually no point in correcting them because the error is relatively harmless. One helpful metaphor: Think of the Internet as a highway, and different services it carries (e.g. the Web and email) as vehicles traveling on that highway.

The process of using the Web goes like this:

1. You enter the name of a server and a page on it, into your client's browser
2. The browser looks up the location of the server using DNS, and sends a request to that address over the Internet.
3. The server receives the request, and replies with information. That information is coded in HTML and CSS – the primary subjects of this book.
4. Your browser receives the HTML and CSS, and *renders* it according to rules we learn in this class.
5. Often you click on a link. The browser understands that this is a request to display another page, and goes back to step 2.

This cycle goes on, with variations we begin to learn in this class, until you exit the browser or turn off your computer.

What about Web 2 and AJAX? These words refer to newer methods which make the Web more lively and interactive – and are beyond the scope of this class. What we learn in this class builds a foundation for learning them later if you need to.

Directories (also called Folders)

What Directories Are: If you are familiar with using directories and already have your documents organized into directories, skip down to "Saving In a Directory." Otherwise, this section is essential – don't move on without understanding it!

This not isn't really about this class, but you need to know. If you place data on an external service, you have less control than you might like about where that data goes. Some services (Facebook is an example) tend to change rules and defaults – and their reasons have much more to do with them making money than you being safe. If you don't want information to go to people who don't care what happens to you, don't put it on a computer you can't control.

One of the most important questions one can ask about computers is, "Where is my data?" It is amazing how quickly computer documents stack up (sort of like papers on your desk). *Back in floppy-disk days we might have one floppy for our correspondence, another for personal finances, another for newsletters, etc. – just as we'd have different file cabinets for different types of files.* Separating things by category cuts the clutter down to reasonable proportions.

Since it can hold so much and you use it for so much of your life hard drive can be a huge clutter problem – just like a closet. But there is a solution: directories. A directory is like a document in that it has a name. But isn't a document: it's a place to put documents. That's why Apple™ calls it a "Folder." It works a lot like a file folder – but with an important difference: it doesn't get all bulgy and malfunction when you fill it up.

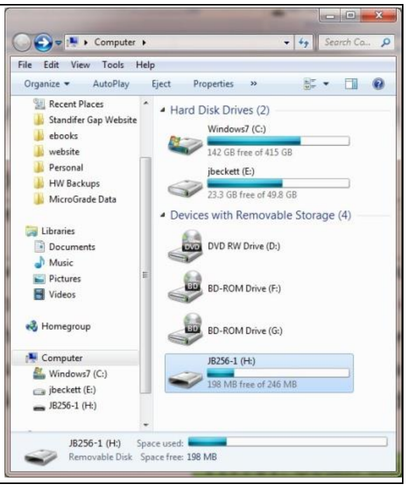
A USB drive may contain directories. That's a good thing, because USB drives can contain thousands of files. It would be a mess to scroll through all of them to find what we want!

Just as important, a directory can contain another directory. Which can contain another directory, and so on.

Navigating Directories: To see what's in a directory, double-click on it.

That will either change the view of the window you're in to a view of what is in the directory, or if you're on the desktop a window will open showing the contents of that directory. *It's like poking your head into the door of a room. No longer do you see what's outside that room, but you see what is in that room.* To change your view back where it was, single-click the back-arrow. *That's like pulling your head back out of the door. You don't see what is inside the room anymore, but what is just outside it.* If you've double-clicked your way down through several levels of directories, you can click your way back out using the back-arrow.

To get to your USB drive, click the Windows button in the lower left-hand corner of the screen, select "Computer", and double-click the USB drive. In the case shown to the right, the USB drive is assigned to H:



Protocols

You can't escape jargon when doing something new, and the Internet is no exception. "Protocol" is a word borrowed from diplomacy, meaning "the rules when communicating in a certain circumstances." Here are some you need to know about at this point:

- **FTP** – File Transfer Protocol, formerly used to upload your files to the server and control what's on the server

- **SFTP** – Secure File Transfer Protocol, an encrypted method of doing FTP. This is our current practice.
- **HTTP**: – HyperText Transfer Protocol, used by browsers to look at Web pages
- **HTTPS**: A variation of HTTP which uses encryption for privacy

HW01: Basic Page
Obtain the template from the class Web page. Change its content to include your name, major, and a hobby. Place it in the HW01 directory of your account on the server, and name it "mypage.htm". Test to make sure it is accessible over the Web using a browser.

Saving In a Directory: Let's get started on our first assignment, a very simple page.

- Bring up a browser, and go to http://computing.southern.edu/jbeckett/WebDev1/HW01_Template.html.txt
- Click anywhere in the page that appears, type control-A, then type control-C. This copies the page to the "clipboard" so you'll have it for the next step. The clipboard is not visible, but like the wind it's important!

Read that paragraph on the left so you understand what a "clipboard" is. It's on the quiz. It's on the final.

- Start Visual Studio Code, click in the document window, and type control-V. This pastes the text you saved on the clipboard, onto the document in the window.
- Save this document where you'll need it. Click File..Save As. A window will appear offering to save it somewhere. It is very unlikely that it's the right place.
- Click the down-arrow at the right side of the window next to "Save in:". Select your login that you use when logging into this computer (probably your name or email username). The window will show several directory icons.
- Double-click the icon named "My Documents." The window will change to show your directories and the files you didn't put in directories.
- Double-click the "Web Homework" directory. The window will show your assignment directories HW01 through HW14.
- Double-click HW01. The window will show the contents of your HW01 assignment (probably empty at this point).
- Enter "mypage.htm" in the "File name:" window and click the Save button.

Why would you save your document before you've done anything to it? Because once you've done this, you can save work as you go along simply by typing control-S (⌘-S on Mac OS). Someone I know *didn't* lose a very critical document when their computer froze the other night, simply because she'd been saving her document along the way.
While we're on this topic, if your document suddenly disappears because you did something wrong (control-A followed by almost any other character is usually the culprit), it's not a good time to type control-S. Think about this *now* so you won't do the wrong thing *then*.

- Replace "This is information about me" with a few sentences telling where you're from, what your major is, and about a hobby you either pursue or would like to. The information you're putting in goes between <p> and </p>.
- Press Control-S to save your document with the new information in it, then exit Visual Studio Code.

Editing functions that use control keys on Windows often use the ⌘Command button on Mac OS. Please note that the click-by-click information on this page is aimed at Windows users. Directory navigation on Mac OS and Linux is similar but not exactly the same.

Uploading

At this point, your assignment is on your computer. Now you need to upload it to the server. But first you'll need to create an account on the homework server. That's easy: Direct your browser at the homework server. Enter your information as requested and click the Submit button. Then check your email in about a minute. You'll get an email telling you two important things:

- How to connect via FTP when using FileZilla
- How to look at your Website using a browser

If you lose this email or forget the password you set up for the homework server, no problem: just re-set your password using the same procedure. Just remember the homework server's address.

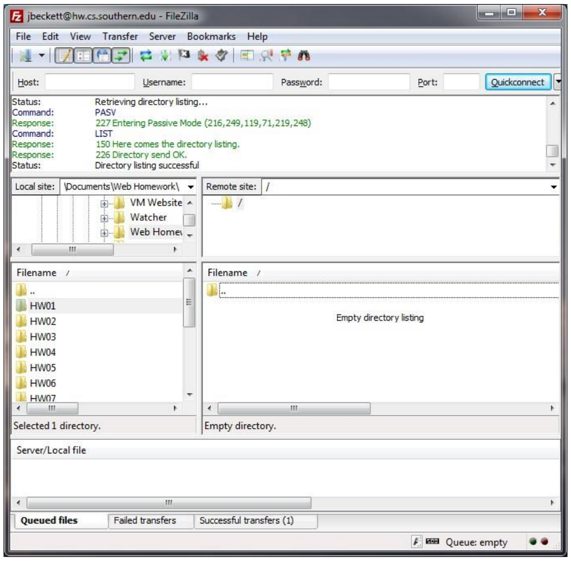
The homework server is at http://hw.cs.southern.edu When you're creating your account it expects your email username (without the "@southern.edu"), your email password, then your desired homework server password (entered twice).

If you have multiple classes that use the Homework Server, you may have another level of directories in use, one for each class. So you'll have to adjust these directions accordingly.

VPN Access: By a policy instituted in the summer of 2023, our server must be behind a firewall to protect it from hackers. You will need to set up a VPN connection between your computer and the server. That process is still under development. You'll need to get that set up with Dr. Scot Anderson in our department before proceeding.

FileZilla Setup: Unless you have some other FTP client such as DreamWeaver, you will probably want to use FileZilla for uploading. Here's how.

1. Enter the parameters for Host and Username (available on the "Find the parameters…" link on the homework server), in the FileZilla entry screen. Also enter the password you gave for the homework server. Leave the Port: blank. Now click the Quickconnect button.
2. If the connection went well, several things will happen faster than you can keep track. The key result is that "Status: Directory listing successful" should appear in the top window. If that doesn't happen, go back and re-set your password on the homework server, wait two minutes, then try again.
3. Navigate the "Local site:" file window to your Web Homework directory. You should see a set of icons for HW01 through HW14. Drag it to the "Remote site:" file window. (If you've done this before, you will be told "The target file already exists" and asked if it's OK to proceed. It's OK.) Your homework is posted!



jbeckett@hw.cs.southern.edu - FileZilla

```
Status:     Retrieving directory listing...
Command:    PASV
Response:   227 Entering Passive Mode (216,249,119,71,219,248)
Command:    LIST
Response:   150 Here comes the directory listing.
Response:   226 Directory send OK.
Status:     Directory listing successful
```

Local site: \Documents\Web Homework\

Remote site: /

Selected 1 directory.

Empty directory.

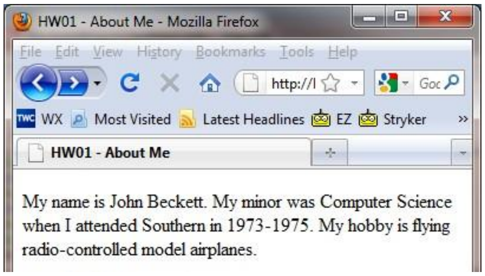
Empty directory listing

Testing the Assignment

This is the most important thing you can do for each assignment: make sure that it works properly when graded!

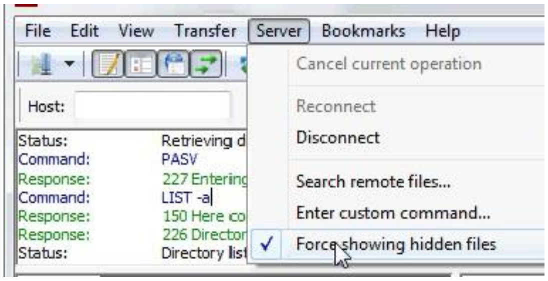
Point your browser at the address given you in the email for viewing the Web pages. If it's a link in the email you received, just click it.

1. You will be asked for your User Name and Password. The user name is everything to the left of the "@" in your email address. The password is the password you gave the homework server.
2. You should see a directory named HW01. Click on it.
3. Then you'll see mypage.htm. Click on it. The result should look somewhat like the picture on the right.


HW01 - About Me - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://l
Goo
WX Most Visited Latest Headlines EZ Stryker
HW01 - About Me

My name is John Beckett. My minor was Computer Science when I attended Southern in 1973-1975. My hobby is flying radio-controlled model airplanes.

If you are running Mac OS, you'll need to make a change to FileZilla's configuration or you won't be able to delete directories you've copied to the server. Here's how: Click on the "Server" tab and select "Force showing hidden files." Easy, isn't it! You only need to do this once – it'll "stick".
It won't hurt to do this if you are using Windows or Linux, but it probably isn't necessary.



```
Status:     Retrieving d
Command:    PASV
Response:   227 Entering
Command:    LIST -a
Response:   150 Here co
Response:   226 Director
Status:     Directory list
```

Lesson 2: Enhancing Text

HTML is a *markup* language – a way of indicating what should be done with information. In case that sounds complicated, let me give an example. If I'm giving my office address to someone over the telephone, I often say among other things "Collegedale – that's one word – Tennessee." They know that "that's one word" doesn't mean that they should write down "that's one word," but that they should write "Collegedale" instead of "College Dale" (two separate words with a space between.) I've given them *meta* information – information about information. HTML uses *tags* to communicate meta information. A tagged item of text has:

- An opening tag, surrounded by "<" and ">"
- The information itself
- A closing tag, surrounded by "</" and ">"

For instance, a title tag looks like this:

```
"<title>This title will appear at the top of my browser window</title>"
```

Organization of an HTML document

There is a certain amount of basic overhead in an HTML document. Even if you are doing very little, it looks complicated. If a cowboy of 1870 were transported magically into current life, he would think we had an awful lot of complicated stuff to handle: a wallet with ID, credit cards, the whole issue of driving. But it's part of what we do and we don't think anything of it. We are going to work through the basic overhead stuff in HTML one bit at a time, so don't let it bother you.

Here's a very basic page that we'll be building on. The line numbers on the left are not part of the page – they are shown so that you can understand which part we're talking about.

```
1  | <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
   | "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
2  | <html xmlns="http://www.w3.org/1999/xhtml">
3  | <head>
4  | <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5  | <title>HW01 - About Me</title>
6  | </head>
7  | <body>
8  | <p>This is information about me.</p>
9  | </body>
10 | </html>
```

Some of the lines don't fit in this illustration, so I've shown them continuing on a second line.

The first thing we do is specify what language we're using, so that the browser will understand how to interpret what is in the document. Americans don't understand this very well because we think there is only one "main" language. But it is not uncommon for Europeans to begin a conversation by agreeing on what language they'll use. That's what is happening here.

- Line 1 is a standard line we'll use for all our assignments. It has technical information about which specific HTML standard we're using.
- Line 2 indicates that this document is in HTML. Line 10 indicates that we are finished with HTML.

Once we're in the language, we have the basic structure.

- Line 3 begins (and line 6 ends) the <head> section. This contains information that won't appear in the browser window.
  - Line 4 indicates the character set we plan to use (which might be very different if we were creating a document in Korean!)
  - Line 5 gives a title to appear in the title bar at the top of the window.

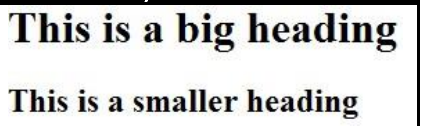
- Line 7 begins (and line 9 ends) the <body> section. This contains information that will appear in the browser window.
  - Line 8 has a very short paragraph – both the tag and the text that will appear in that paragraph.

Incantation: A computer term for code that you don't really need to understand, but which does something you need done. Think of it as a phrase out of a tourist guide to a foreign country, and you have the idea. Just as you must be extra-careful to pronounce what's in the tourist guide correctly, however, punctuation and syntax are critical.

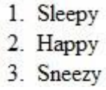
Body Tags

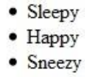
All we will be changing in the <head> section is the <title> text. For the rest of this class, we will focus on the <body> section. Here's an overview of the tags we will cover in this lesson, so you'll have them all in one place.

Tag	Purpose	Example	Rendered by Browser

<h*n*>	Create a heading. This is placed on a separate line. *n* may be 1 to 6.	`<h1>This is a big heading</h1>`
		`<h2>This is a smaller heading</h2>`
			 **This is a big heading**
			This is a smaller heading

<p>	Paragraph of text.	`<p>Now is the time for all good people to come to the aid of their country.</p>`	Now is the time for all good people to come to the aid of their country.
<b>	Boldface. In our example, we show a single word boldfaced.	`<p>Now is the time for all <b>good</b> people to come to the aid of their country.</p>`	Now is the time for all **good** people to come to the aid of their country.
<i>	Italics.	`<p>Now is the time for all <i>good</i> people to come to the aid of their country.</p>`	Now is the time for all *good* people to come to the aid of their country.

<ol>
<li>
	Ordered (numbered) list. Enclose items in <li> tags.	`<ol><li>Sleepy</li>`
		`<li>Happy</li>`
		`<li>Sneezy</li></ol>`
			 1. Sleepy
			2. Happy
			3. Sneezy

<ul>
<li>
	Unordered (bulleted) list.	`<ul><li>Sleepy</li>`
		`<li>Happy</li><li>sneezy></ul>`
			 • Sleepy
			• Happy
			• Sneezy

<hr>	Create a "Horizontal Rule"	`<hr />`	
<br>	Create a line break	`Happy<br /> Birthday`	Happy<br>Birthday

Now we'll give the details for each of these tags.

<p>

The paragraph tag encloses a block of text that should be handled as a paragraph. It will start a new line, and have the end of a line at its end. Lines will automatically be broken up ("wrapped") to fill the width available.

Tags in XHTML must be done in a certain order: Whatever order you got into something, you must back out in reverse order. Think of using string to keep track of your exit route in a cave: If you meet the string, you risk going in circles if you follow the piece of string you just met. To be safe, you must go back the way you come.

<b>

The tagged text is emphasized by bold-facing. Note that some fonts don't look much (if any) different when bold-faced.

<i>

The tagged text is emphasized by italics.

<ol>

An Ordered List is a compound tag structure. It shows an ordering indication (by default a number beginning with 1). You write an `<ol>` tag defining the overall list, and `<li>` tags enclosing each item in the list.

An ordered list can use something other than 1..2..3 for its ordering indicator. For example, you could write:
`<ol type="a">`
Here are the options:
`1` – 1,2,3,4…
`a` – a,b,c,d…
`A` – A,B,C,D…
`i` – i,ii,iii,iv…
`I` – I,II,III,IV…

<ul>

An Unordered List is a compound tag structure. It shows a separator (by default a round bullet). You write a `<ul>` tag defining the overall list, and `<li>` tags enclosing each item in the list.

An unordered list can use something other than "•" for its item indicator. Use the "type" property. For example, you could write:
`<ul type="circle">` to create a list using open circles for bullets.
Here are some options:
type="disc" - *Displays list items in bullets*
type="circle" - *Displays list items in circles*
type="square" - *Displays list items in squares*

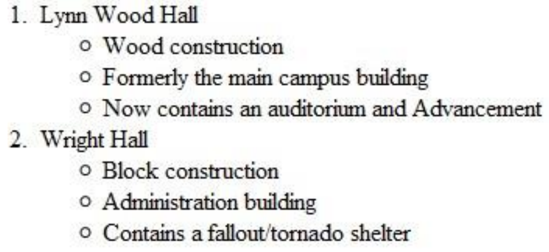
Nested Lists

Ordered and unordered lists may be nested, much like outline form you already know about. Remember: Whichever way you go in, you have to go back out in the opposite order. The best way to illustrate this is with an example: A numbered list, with bulleted items within each item of the list.

Code

```
<ol>
  <li>Lynn Wood Hall<ul>
      <li>Wood construction</li>
         <li>Formerly the main campus building</li>
         <li>Now contains an auditorium and
Advancement</li>
         </ul>
  </li>
  <li>Wright Hall<ul>
      <li>Block construction</li>
         <li>Administration building</li>
         <li>Contains a fallout/tornado shelter</li>
         </ul>
  </li>
</ol>
```

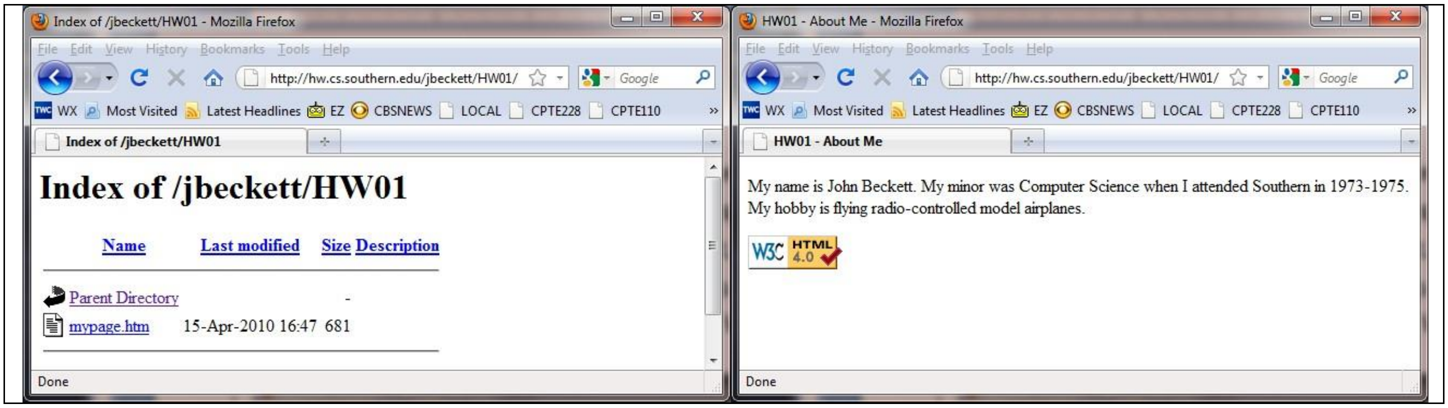
Result



1. Lynn Wood Hall
   - Wood construction
   - Formerly the main campus building
   - Now contains an auditorium and Advancement
2. Wright Hall
   - Block construction
   - Administration building
   - Contains a fallout/tornado shelter

Directory Index

Each Web site's server has a configuration setting that allows you to create a page that will be displayed when a visitor points their browser at a site or directory without specifying a file name. This configuration setting is a list of file names that will be checked by the server to see if they exist. If one of those files exists, it will be used displayed instead of a list of the files. Sound confusing? An example might help. On the left is the appearance of my Web site after completing HW01. On the right is shown the appearance of my Web site if I named the first assignment "index.htm" instead of "mypage.htm".



Index of /jbeckett/HW01

Name	Last modified	Size	Description
Parent Directory		-	
mypage.htm	15-Apr-2010 16:47	681	

My name is John Beckett. My minor was Computer Science when I attended Southern in 1973-1975. My hobby is flying radio-controlled model airplanes.

Index of /jbeckett/HW01 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://hw.cs.southern.edu/jbeckett/HW01/
Google
WX Most Visited Latest Headlines EZ CBSNEWS LOCAL CPTE228 CPTE110
Index of /jbeckett/HW01
Done
HW01 - About Me - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://hw.cs.southern.edu/jbeckett/HW01/
Google
WX Most Visited Latest Headlines EZ CBSNEWS LOCAL CPTE228 CPTE110
HW01 - About Me
W3C HTML 4.0
Done

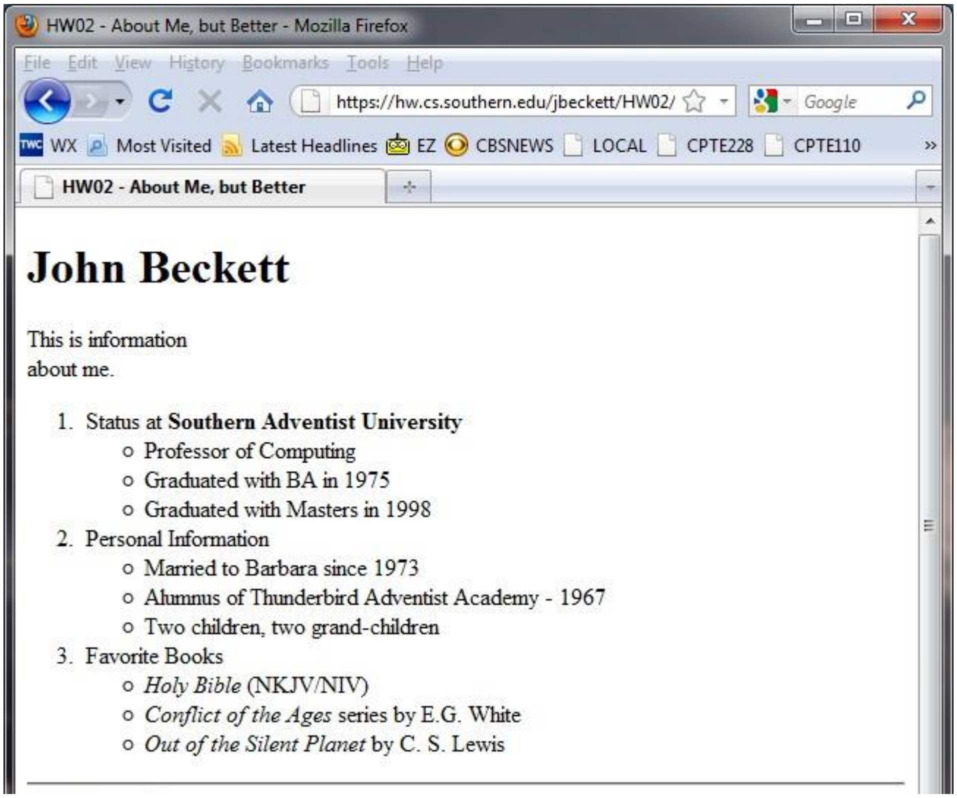
What's interesting is that the URL is exactly the same: http://hw.cs.southern.edu/jbeckett/HW01/. The only difference is the name of my project file. In the first case the Web server didn't find any file it knows as an index (index.htm or index.html), so it created an index file with a link to the file it *did* find. In the second case there was a file named index.htm – which the server displayed immediately. And since it was displayed like this, the visitor has no idea what the document's actual file name is.

Techies may wonder what our list of index files is. It's pretty typical, and this is an exact copy from our server:

```
    DirectoryIndex index.php index.html index.htm index.shtml
```

Assignment

For this lesson, we will create a better-looking page (which might be about yourself). Here’s my example.


HW02 - About Me, but Better - Mozilla Firefox
File Edit View History Bookmarks Tools Help
https://hw.cs.southern.edu/jbeckett/HW02/
Google
WX Most Visited Latest Headlines EZ CBSNEWS LOCAL CPTE228 CPTE110
HW02 - About Me, but Better

John Beckett

This is information
about me.

1. Status at **Southern Adventist University**
   - Professor of Computing
   - Graduated with BA in 1975
   - Graduated with Masters in 1998
2. Personal Information
   - Married to Barbara since 1973
   - Alumnus of Thunderbird Adventist Academy - 1967
   - Two children, two grand-children
3. Favorite Books
   - *Holy Bible* (NKJV/NIV)
   - *Conflict of the Ages* series by E.G. White
   - *Out of the Silent Planet* by C. S. Lewis

And here’s the specification.

HW02: Text Enhancements

- Use the same template you used for HW01. This assignment will be placed in a directory named HW02.
- The file name for your document must be either index.html or index.htm, so that no directory index will be shown (as on the left above), but the visitor will be shown the assignment immediately when they browse to your Website and select HW02.
- Change the page title to “About Me, but Better.”
- Make a <h1> type heading
- Make a <p> heading as shown with a <br /> tag between “information” and “about me”
- Use nested <ol> and <ul> for three categories of information, with three bullets in each. The exact categories and bullets aren’t important for this assignment – the structure is. In fact, this doesn’t even have to be about you if you’d rather use some other subject.
- Add a horizontal rule after your personal information.

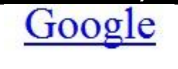
Place your page, naming it “index.htm”, in the HW02 directory of your account on the server. Test to make sure it works correctly over the Web using a browser.

Lesson 3 – Links

You have undoubtedly used hyperlinks (called “links” in this book). Clicking on a hyperlink causes the document you’re seeing to be replaced by another document – perhaps a specific place in that document.

It’s done with the <a> tag, which means “anchor.” If you wrap an object (words or an image) with an <a>, clicking on that object causes the viewed document to change to the document cited.

A Simple Case

Code	Rendered by Browser
`<a href="http://google.com">Google</a>`	 Google

The word “Google” appears in blue and is underlined. Click and you’re at Google’s Website. To return you’d have to click a back-arrow. Let’s break down the components of this code so we can expand on the idea.

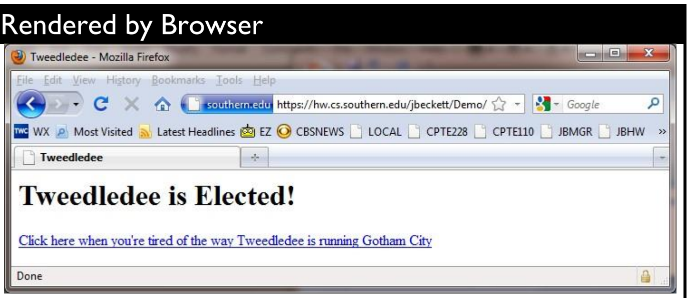
• `<a`	This is an anchor tag
• `href=`	When this object is clicked do a hypertext reference to…
• `"http://google.com"`	Here’s where we are going
• `Google`	The text to display for this link
• `</a>`	`Closing the tag`

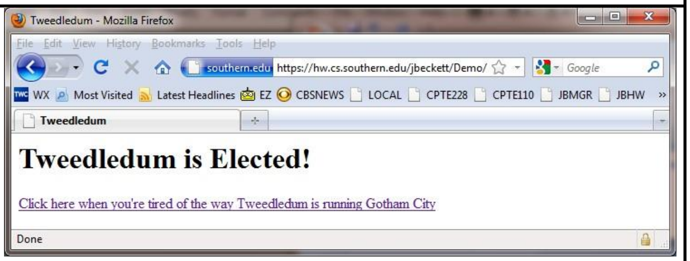
Now that was easy, wasn’t it? Most of this lesson has to do with forming the information inside quotes in that href entry.

Linking To Another Page On the Same Site

In the examples above, we've used a location that wasn't part of our site. If we want to specify another page located in the same directory with our original page on the server, it's even easier. Here is code for two pages that allow a visitor to click back and forth between them.

File	Code	Rendered by Browser

tweedledee.htm	`<html><head>`
	`<title>Tweedledee</title></head>`
	`<body>`
	`  <h1>Tweedledee is Elected!</h1>`
	`  <a href="tweedledum.htm">`
	`    Click here when you're tired of`
	`    the way Tweedledee is running`
	`    Gotham City</a>`
	`</body></html>`
		
		Tweedledee - Mozilla Firefox
		https://hw.cs.southern.edu/jbeckett/Demo/
		Tweedledee
		Tweedledee is Elected!
		Click here when you're tired of the way Tweedledee is running Gotham City
		Done

tweedledum.htm	`<html><head>`
	`<title>Tweedledum</title></head>`
	`<body>`
	`  <h1>Tweedledum is Elected!</h1>`
	`  <a href="tweedledee.htm">`
	`  Click here when you're tired of`
	`  the way Tweedledum is running`
	`  Gotham City</a>`
	`</body></html>`
		
		Tweedledum - Mozilla Firefox
		https://hw.cs.southern.edu/jbeckett/Demo/
		Tweedledum
		Tweedledum is Elected!
		Click here when you're tired of the way Tweedledum is running Gotham City
		Done

On the first page, there is an anchor tag that switches control to the second page. On the second page, there is an anchor tag that switches control to the first tag.

Using nothing but what we know so far, we could easily implement a Paper Rock Scissors game using multiple HTML documents. Here's the code, with filenames on the left:

PRSBegin.htm	<html><head><title>Paper, Rock, Scissors - New Game</title> </head><body> <p>Player 1's turn</p> <p>Select:</p> <ul> <li><a href="P1P.htm">Paper</a></li> <li><a href="P1R.htm">Rock</a></li> <li><a href="P1S.htm">Scissors</a></li> </ul> </body> </html>	**P1P.htm**	<html><head><title>Player 1 selected Paper</title> </head><body> <p>Player 2's turn</p> <p>Select:</p> <ul> <li><a href="PRSTie.htm">Paper</a></li> <li><a href="PRSLose.htm">Rock</a></li> <li><a href="PRSWin.htm">Scissors</a></li> </ul> </body> </html>
P1R.htm	<html><head><title>Paper, Rock, Scissors - New Game</title> </head><body> <p>Player 2's turn</p> <p>Select:</p> <ul> <li><a href="PRSWin.htm">Paper</a></li> <li><a href="PRSTie.htm">Rock</a></li> <li><a href="PRSLose.htm">Scissors</a></li> </ul> </body> </html>	**P1S.htm**	<html><head> <title>Paper, Rock, Scissors - New Game</title> </head><body> <p>Player 2's turn</p> <p>Select:</p> <ul> <li><a href="PRSLose.htm">Paper</a></li> <li><a href="PRSWin.htm">Rock</a></li> <li><a href="PRSTie.htm">Scissors</a></li> </ul> </body> </html>
PRSLose.htm	<html><head><title>Player 2 loses</title> </head><body> <h1>Player 2 lost!</h1> <p><a href="PRSBegin.htm">New game</a></p> </body></html>	**PRSTie.htm**	<html><head><title>Tie Game</title> </head><body> <h1>Tie!</h1> <p><a href="PRSBegin.htm">New game</a></p> </body></html>
		PRSWin.htm	<html><head><title>Player 2 wins</title> </head><body> <h1>Player 2 won!</h1> <p><a href="PRSBegin.htm">New game</a></p> </body></html>

This game actually works. It has one disadvantage: By paying attention, the second player could easily figure out what the first player had done and select a winning turn every time. But not to worry, this is just a demonstration.

Types of URLs

Here's something very important we should notice at this point: We didn't include the `http://...` in the hyperlink reference. There are actually several ways you can code the href. I'll give you the list first, then work through it.

Type Example Explanation

URL Type	Example	Explanation
Absolute	http://www.cs.southern.edu	Go to a specific server.
Absolute with directory	http://www.cs.southern.edu/jbeckett	Go to a specific directory on a specific server
Directory	/scot	Goes to a specific directory on the same server as the current page. For instance: If I used this url in a link on one of the pages on my personal Web site, it would send the user to Dr. Scot Anderson's personal Web site
Relative	tweedledum.htm	Goes to a file in the same directory as the current page.
Another protocol	ftp://hw.cs.southern.edu/jbeckett	(This is an absolute url.) Use the FTP server instead of the Web server
Personal Directory	http://www.cs.southern.edu/~jbeckett	Some sites use "tilde access" to connect visitors to individual Web sites.
Target within the same file	#service	This sends the browser to the same file, to a place marked with id="service".

- No preamble of any kind – just look for a file by this name in the same directory as the page we're on. We call this a **relative** link.
- Begin with a protocol name (`https://` is a protocol) – Go somewhere on the Internet. We call this an **absolute** link.

It would be possible to do everything on the World Wide Web using absolute links. And if nobody ever moved any pages around, that would be just fine. But one of the things we can count on is change. Relative links keep working if the site is moved to a different server or a different place on the same server. So we use relative links whenever possible.

What's the difference between http:// and https://? They do very much the same thing except that when you're using https:// the information going between your client and the server is encrypted – meaning that people "sniffing" it (usually over WiFi) can't read it. Until we get to form handling, I'll be treating these two as interchangeable.

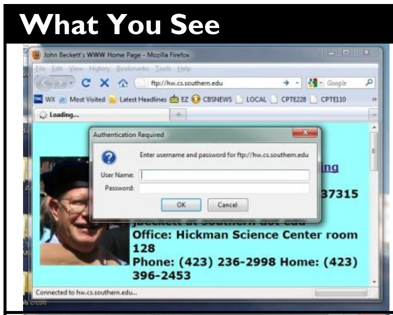
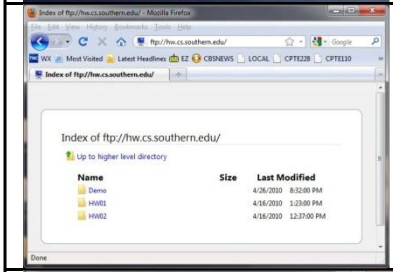
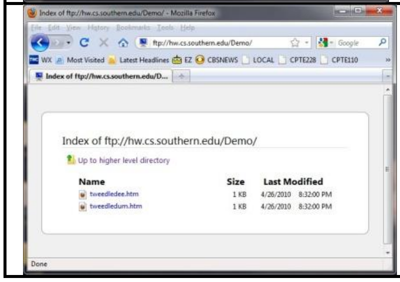
There are other types of URLs:

- Start with a slash – Use the same server, but go to an absolute location on that server. So far I haven't figured out a use for this type, so you won't see much of it in the rest of this course. But sometimes people use it and you need to know what it means.
- Start with two dots and a slash– Goes one level "up" in the directory structure.

What about backward and forward slashes? Microsoft Windows tends to use backward slashes ("\") as directory delimiters. The rest of the world uses forward slashes ("/"). This is because of confusion back in DOS days (a controversy that is fascinating to techies but not important to us here), and we can expect the difference to continue. Web servers including Microsoft's Internet Information Server use forward slashes. So if you're using Windows this assignment will have a file in your homework directory that is at HW03\quotes\index.htm, but on the server it will be HW03/quotes/index.htm. Some Web servers, browsers, and even Windows 7's command window, will automatically translate slashes into whichever mode is appropriate for the context.

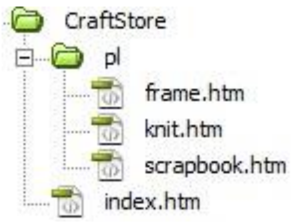
Other Protocols

You will occasionally see a different protocol used. For example, you can use ftp to access your homework account on our server. In this case I point my browser at `ftp://hw.cs.southern.edu`.

What You See	Explanation
 John Beckett's WWW Home Page - Mozilla Firefox ftp://hw.cs.southern.edu Authentication Required Enter username and password for ftp://hw.cs.southern.edu User Name: Password: Office: Hickman Science Center room 128 Phone: (423) 236-2998 Home: (423) 396-2453 Connected to hw.cs.southern.edu... File Edit View History Bookmarks Tools Help WX Most Visited Latest Headlines EZ CBSNEWS LOCAL CPTE228 CPTE110 Loading... OK Cancel Google	The browser asks the server for ftp access. The server does not have any such access available without a username and password, but given those it would know where to go. So it asks for authentication. So I type in "jbeckett" and my homework server password.
 Index of ftp://hw.cs.southern.edu/ - Mozilla Firefox ftp://hw.cs.southern.edu/ Index of ftp://hw.cs.southern.edu/ Up to higher level directory Name Size Last Modified Demo 4/26/2010 8:32:00 PM HW01 4/16/2010 1:23:00 PM HW02 4/16/2010 12:37:00 PM Done File Edit View History Bookmarks Tools Help WX Most Visited Latest Headlines EZ CBSNEWS LOCAL CPTE228 CPTE110 Google	The server sends a directory of the files available. I click on "Demo".
 Index of ftp://hw.cs.southern.edu/Demo/ - Mozilla Firefox ftp://hw.cs.southern.edu/Demo/ Index of ftp://hw.cs.southern.edu/Demo/ Up to higher level directory Name Size Last Modified tweedledee.htm 1 KB 4/26/2010 8:32:00 PM tweedledum.htm 1 KB 4/26/2010 8:32:00 PM Done File Edit View History Bookmarks Tools Help WX Most Visited Latest Headlines EZ CBSNEWS LOCAL CPTE228 CPTE110 Index of ftp://hw.cs.southern.edu/D... Google	The server sends another directory, this time showing all the files in the demo directory.

How is this useful? It lets us see what is in our directory without automatically activating the index file as we saw in the second lesson. And it demonstrates that if we have an FTP server, we can get things from it using a browser. In fact, this is how Southern Adventist University first implemented its WWW page back in 1994 – before we had Web server software installed. Usually a browser isn't best for FTP access, but occasionally it's the best way – especially if you need to obtain a file using FTP and you're on a computer that doesn't have FileZilla or similar software already installed.

Pointing to other directories



```
CraftStore
  pl
    frame.htm
    knit.htm
    scrapbook.htm
  index.htm
```

Web sites could theoretically be set up “flat” – with all documents and other materials in the same directory. As we mentioned back in lesson 1, this gets cumbersome. So it is common practice to use directories to organize our materials. For instance, we could have a directory with a set of documents that are price lists for our various product lines. One could cover scrap-booking supplies, another might be knitting, and yet another picture frames. I’ve used Dreamweaver to illustrate this structure in the diagram to the right, as we work through an example of linking to files in other. The code for all four pages is shown below, and is implemented on the support Website for this book. I’ve omitted some preamble information for brevity.

index.htm	`<html><head>`
	`<title>Gabriel's Hobby Store</title>`
	`</head><body>`
	`<p>Welcome to Gabriel's Hobby Store online.`
	`Click below for prices</p>`
	`<p><a href="pl/scrapbook.htm">Scrap-`
	`Booking</a><br />`
	`<a href="pl/knit.htm">Knitting</a><br />`
	`<a href="pl/frame.htm">Picture frames</a></p>`
	`</body></html>`
		frame.htm	`<html><head>`
			`<title>Frames at Gabriel's</title>`
			`</head>`
			`<body><h1>Frame Prices at Gabriel's</h1>`
			`<p>Stock Frames: $5.00 per diagonal inch<br />`
			`Custom Frames: $3.00 per inch of border</p>`
			`<a href="../index.htm">Back</a></body>`
			`</html>`

knit.htm	`<html><head>`
	`<title>Knitting at Gabriel's</title>`
	`</head><body>`
	`<h1>Knitting prices at Gabriel's</h1>`
	`<p>We have a wide variety of polyester`
	`knitting yarn, all at $1.99 per skein</p>`
	`<a href="../index.htm">Back</a>`
	`</body></html>`
		scrapbook.htm	`<html> <head>`
			`<title>Scrap-Booking at Gabriel's</title>`
			`</head><body>`
			`<h1>Scrap-Booking prices</h1>`
			`<p>Our competition is our catalog -`
			`we guarantee no price higher than theirs, and often`
			`charge 10% less</p>`
			`<a href="../index.htm">Back</a>`
			`</body></html>`

Now let’s explain the directory references in this code.

Child directory

index.htm is in the primary (usually called “root”) directory for the store. So if we wish to reference the page on knitting prices, we must mention the directory the price lists are in – “pl”. Each of the three pages in the pl directory is addresses by “`pl/`*`something`*`.htm`”. This is a **relative** reference – meaning that it is relative to the current document.

Pointing To the Parent directory

In each of the three pages in the pl directory, there is a reference to the index page. This is indicated by the notation “`../index.htm`”. The two dots mean “go up a directory.”

Grand-children and grand-parents

It is possible for a directory to have another directory. If Gabriel’s and a store named “HomeStuff” were both owned by the same corporation, for instance, we might have a directory above “CraftStore” which would have “CraftStore” and “Home” in it, each with its own “pl” directory. So if we had a special on frames and wished to highlight this in the index for the corporation, we could code:

```
<a href="CraftStore/pl/frames.htm">Special on Frames at Gabriel’s!</a>
```

And likewise, if we wanted a link on the frames price page for Gabriel’s that takes the visitor to the master page for both stores, we could code:

```
<a href="../../index.htm”>Parent Corporate site</a>
```

On-page

Sometimes you have a page that doesn’t fit on a screen (or you don’t know that your site visitors will all have large screens). You can set up links that go to different parts of the same page instead of different pages. There are two parts to this process:

1. Since it’s bad form to shoot before we know what we’re shooting at, we begin by identifying the target. This is done by using the anchor tag with an “id” attribute instead of “href.” Like this:

   ```
   <a id="details" />
   ```

   This code does not cause anything to appear on the visitor’s screen.
2. Now for the code that will move the screen to that target:

   ```
   <a href="#details">More information</a>
   ```

   When the words “More information” are clicked, the visitor’s screen will shift as necessary to show the location identified with the first tag. Note that if that target is already on-screen, nothing will happen.

There is a demo of this method on the support Website.

Location on another page

You can combine the two techniques, so that you can send the visitor to a specific place on another page. Let’s say you have a page named “products.htm” and another page named “pricelist.htm”. You could identify sections of “pricelist.htm” with anchor using the “id=” attribute, and go to them from links on “products.htm.” We’ll show this with code snippets rather than entire pages.

products.htm	pricelist.htm
`<a href="pricelist.htm#lcd">LCD panel Prices</a>`	`<a id="lcd">LCD panels <br />15" - $40<br />17" - $90`

URL encoding of variables

It is also possible to send information to back-end processing programs by encoding it on the URL line. For example:

```
<a href="process.php?mode=4&firstname=John&lastname=Beckett">Post this entry</a>
```

The question mark indicates that we have information to send to the next page. Each item of information has two parts: a variable name, and the value to send for that variable. As we'll see in lesson 9, this is very useful for handling forms people fill out. But it can also be useful for sending instructions. For instance, you could have two links that send the site visitor to your employee directory. One of them could send a code indicating that the directory should be sorted by department, and the other link could send a code indicating that a full directory sorted alphabetically by name is required. The same back-end program could handle both queries, and base its handling on a url-encoded variable.

If you're using a search engine, it often works this way. For instance, what if I look for myself on Google. If I type "John Beckett" into the Google search window, it sends me to this URL:

http://www.google.com/search?q=John+Beckett&ie=utf-8&oe=utf-8&aq=t&rls=org.mozilla:en-US:official&client=firefox-a

There is a special advantage: you can pass this link to someone else and they'll get the same or similar results. You can actually copy this line if you're reading this book online, paste it into a browser, and see what comes up.a

Link to same or different window

By default, href links send the site visitor to a new location by re-loading the window with whatever is specified. But what if you don't want to do this? You can specify that a link is to be handled by creating a new window or tab, like this:

```
<a href="http://google.com" target="_blank">Create a Google window</a>
```

The target attribute indicates that the new page should be loaded somewhere other than the current page, and a new window (usually a new tab in recent browsers) will be created to hold this page.

HW03: Links
This assignment will contain multiple files in multiple directories.

Directory	File	Contents	Links
HW03			
	index.htm	Adapt HW02 index	Menu bar for all pages in HW03 Quotes Hobby
	hobby.htm	Description of hobby	Menu bar for all pages in HW03 Two external sites – one opening in this window, the other opening in a separate window
HW03/quotes			
	index.htm		Menu bar for all pages in HW03 Links to several places on egw.htm Bible.htm
	egw.htm	Plenty of text	Menu bar for all pages in HW03 On-page links to various sections – with a links bar at the beginning of each section so the visitor doesn't have to click the "back arrow" button
	Bible.htm	Several verses	Menu bar for all pages in HW03 Links to alternate translations or languages

Suggested procedure:

1. Copy your index.htm page from HW02 into the HW03 directory.
   a. Change its title and contents as appropriate for this assignment (see example below).
   b. Add a menu bar at the top (right under the <h1> heading. Here's mine:

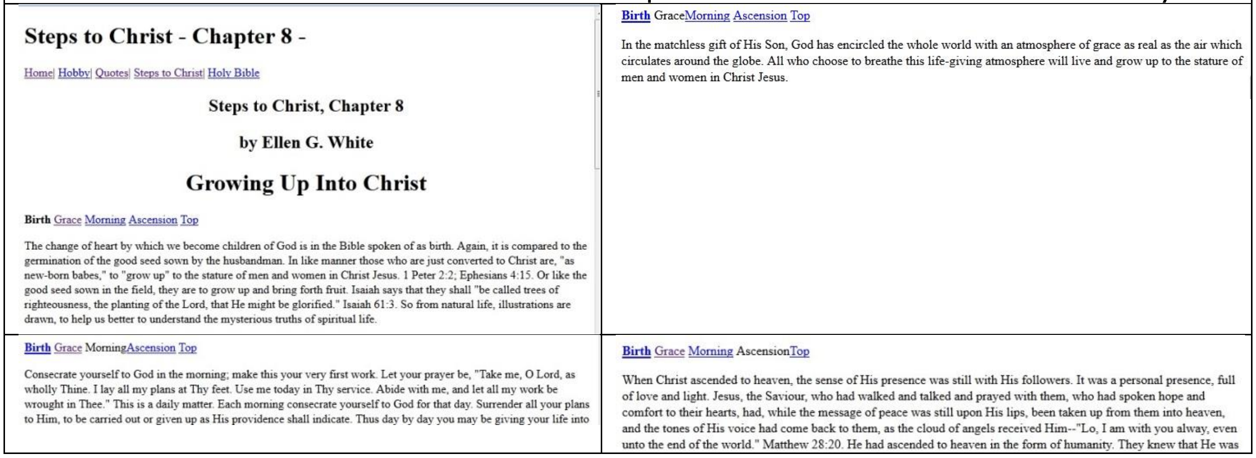
```
<p>
  <a href="index.htm">Home</a>|
  <a href="hobby.htm">Hobby</a>|
  <a href="quotes/index.htm">Quotes</a>|
  <a href="quotes/egw.htm">Steps to Christ</a>|
  <a href="quotes/Bible.htm">Holy Bible</a>
</p>
```

2. Copy the index.htm page you just made, to a new file named hobby.htm in the same directory.
   a. Change its title and contents as appropriate for this assignment.
   b. You won't have to modify the menu bar at the top.
3. Create a directory under HW03 called "quotes". The other three pages in this assignment will be located in this directory.
4. Copy HW03/index.htm to HW03/quotes/index.htm.
   a. Change its title and contents as appropriate for this assignment.
   b. You'll have to modify the menu bar because you're in a different directory.
      i. The "Home" and "Hobby" pages are now in the directory above (remember: that's done with "`../`")
      ii. The "Quotes", "Steps to Christ", and "Holy Bible" pages are now in the same directory so you'll be removing the "`quotes/`" from those references.

5. Copy that index.htm file you just made to “egw.htm” and “Bible.htm”. Note: I’ve capitalized the word “Bible” in the filename, a common practice among Christians. In HTML this means you’ll have to capitalize it the same way in references. This assignment will work just fine with that name capitalized or not, but you must be consistent.
6. Modify the title and contents for egw.htm and Bible.htm as needed.
    a. You’ll be putting in some on-page links in egw.htm. Here’s my first one. Since this identifies “Birth”, it assigns it a name, and I’ve boldfaced the word. The other entries are links elsewhere in the document. You can take this line and modify it for the other locations.

```
<b><a name="Birth" />Birth</b>
   <a href="#Grace">Grace</a>
   <a href="#Morning">Morning</a>
   <a href="#Ascension">Ascension</a>
   <a href="#Top">Top</a><br />
```

These links will take the site visitor to four different places in the document. Here is where they are:



Steps to Christ - Chapter 8 -

Home| Hobby| Quotes| Steps to Christ| Holy Bible

Steps to Christ, Chapter 8

by Ellen G. White

Growing Up Into Christ

Birth Grace Morning Ascension Top

The change of heart by which we become children of God is in the Bible spoken of as birth. Again, it is compared to the germination of the good seed sown by the husbandman. In like manner those who are just converted to Christ are, "as new-born babes," to "grow up" to the stature of men and women in Christ Jesus. 1 Peter 2:2; Ephesians 4:15. Or like the good seed sown in the field, they are to grow up and bring forth fruit. Isaiah says that they shall "be called trees of righteousness, the planting of the Lord, that He might be glorified." Isaiah 61:3. So from natural life, illustrations are drawn, to help us better to understand the mysterious truths of spiritual life.

Birth GraceMorning Ascension Top

In the matchless gift of His Son, God has encircled the whole world with an atmosphere of grace as real as the air which circulates around the globe. All who choose to breathe this life-giving atmosphere will live and grow up to the stature of men and women in Christ Jesus.

Birth Grace MorningAscension Top

Consecrate yourself to God in the morning; make this your very first work. Let your prayer be, "Take me, O Lord, as wholly Thine. I lay all my plans at Thy feet. Use me today in Thy service. Abide with me, and let all my work be wrought in Thee." This is a daily matter. Each morning consecrate yourself to God for that day. Surrender all your plans to Him, to be carried out or given up as His providence shall indicate. Thus day by day you may be giving your life into

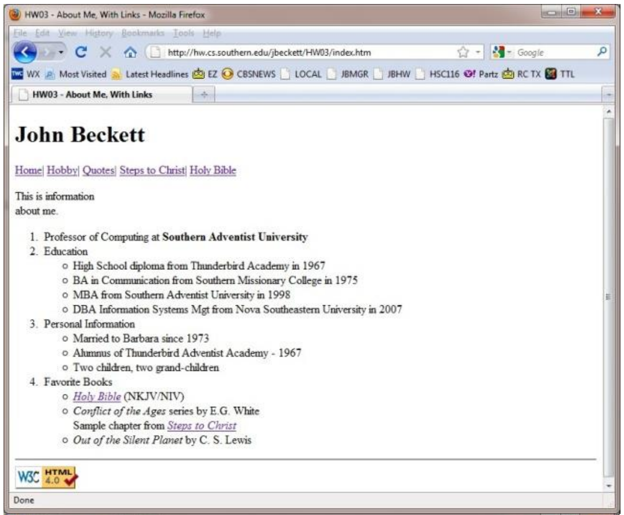
Birth Grace Morning AscensionTop

When Christ ascended to heaven, the sense of His presence was still with His followers. It was a personal presence, full of love and light. Jesus, the Saviour, who had walked and talked and prayed with them, who had spoken hope and comfort to their hearts, had, while the message of peace was still upon His lips, been taken up from them into heaven, and the tones of His voice had come back to them, as the cloud of angels received Him--"Lo, I am with you alway, even unto the end of the world." Matthew 28:20. He had ascended to heaven in the form of humanity. They knew that He was

7. Go back over your assignment, making sure you’ve set up all the links that need to be created (see the green box above). Any “404” (page not found) is a fail!
8. Upload your assignment.
9. Test your assignment to make sure all the links work on the server using http:// access with a browser. Again, any “404” is a fail.

Here are displays of pages in this assignment.

HW03/index.htm


HW03 - About Me, With Links - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://hw.cs.southern.edu/jbeckett/HW03/index.htm
Google
WX Most Visited Latest Headlines EZ CBSNEWS LOCAL JBMGR JBHW HSC116 Partz RC TX TTL
HW03 - About Me, With Links
W3C HTML 4.0
Done

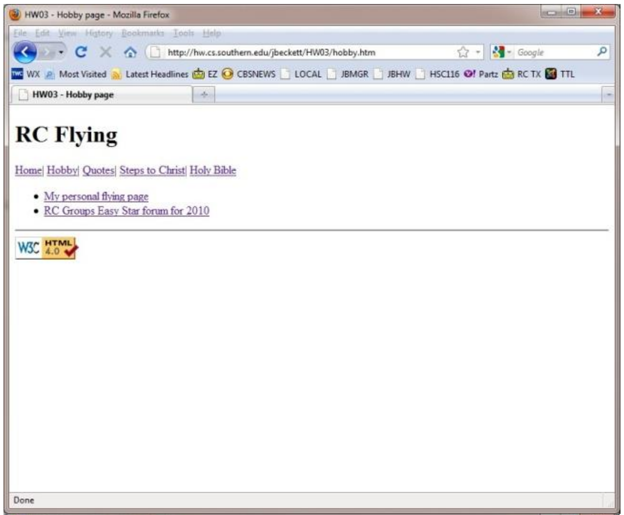
John Beckett

Home| Hobby| Quotes| Steps to Christ| Holy Bible

This is information
about me.

1. Professor of Computing at **Southern Adventist University**
2. Education
   - High School diploma from Thunderbird Academy in 1967
   - BA in Communication from Southern Missionary College in 1975
   - MBA from Southern Adventist University in 1998
   - DBA Information Systems Mgt from Nova Southeastern University in 2007
3. Personal Information
   - Married to Barbara since 1973
   - Alumnus of Thunderbird Adventist Academy - 1967
   - Two children, two grand-children
4. Favorite Books
   - *Holy Bible* (NKJV/NIV)
   - *Conflict of the Ages* series by E.G. White
     Sample chapter from *Steps to Christ*
   - *Out of the Silent Planet* by C. S. Lewis

HW03/hobby.htm

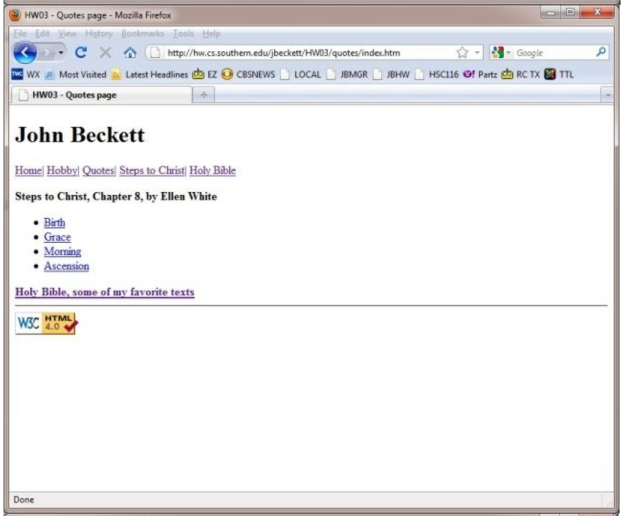

HW03 - Hobby page - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://hw.cs.southern.edu/jbeckett/HW03/hobby.htm
Google
WX Most Visited Latest Headlines EZ CBSNEWS LOCAL JBMGR JBHW HSC116 Partz RC TX TTL
HW03 - Hobby page
W3C HTML 4.0
Done

RC Flying

Home| Hobby| Quotes| Steps to Christ| Holy Bible

- My personal flying page
- RC Groups Easy Star forum for 2010

HW03/quotes/index.htm


HW03 - Quotes page - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://hw.cs.southern.edu/jbeckett/HW03/quotes/index.htm
Google
WX Most Visited Latest Headlines EZ CBSNEWS LOCAL JBMGR JBHW HSC116 Partz RC TX TTL
HW03 - Quotes page
W3C HTML 4.0
Done

John Beckett

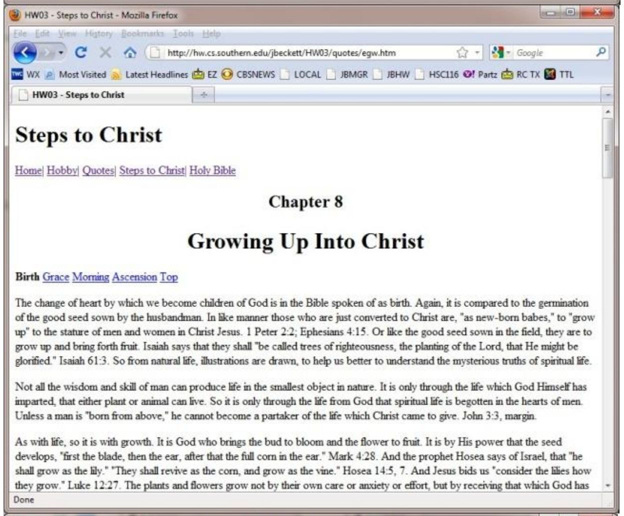
Home| Hobby| Quotes| Steps to Christ| Holy Bible

Steps to Christ, Chapter 8, by Ellen White

- Birth
- Grace
- Morning
- Ascension

Holy Bible, some of my favorite texts

HW03/quotes/etw.htm


HW03 - Steps to Christ - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://hw.cs.southern.edu/jbeckett/HW03/quotes/egw.htm
Google
WX Most Visited Latest Headlines EZ CBSNEWS LOCAL JBMGR JBHW HSC116 Partz RC TX TTL
HW03 - Steps to Christ
Done

Steps to Christ

Home| Hobby| Quotes| Steps to Christ| Holy Bible

Chapter 8

Growing Up Into Christ

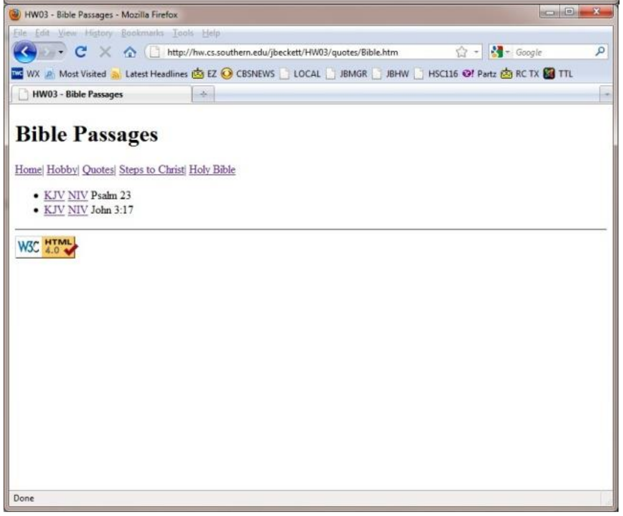
Birth Grace Morning Ascension Top

The change of heart by which we become children of God is in the Bible spoken of as birth. Again, it is compared to the germination of the good seed sown by the husbandman. In like manner those who are just converted to Christ are, "as new-born babes," to "grow up" to the stature of men and women in Christ Jesus. 1 Peter 2:2; Ephesians 4:15. Or like the good seed sown in the field, they are to grow up and bring forth fruit. Isaiah says that they shall "be called trees of righteousness, the planting of the Lord, that He might be glorified." Isaiah 61:3. So from natural life, illustrations are drawn, to help us better to understand the mysterious truths of spiritual life.

Not all the wisdom and skill of man can produce life in the smallest object in nature. It is only through the life which God Himself has imparted, that either plant or animal can live. So it is only through the life from God that spiritual life is begotten in the hearts of men. Unless a man is "born from above," he cannot become a partaker of the life which Christ came to give. John 3:3, margin.

As with life, so it is with growth. It is God who brings the bud to bloom and the flower to fruit. It is by His power that the seed develops, "first the blade, then the ear, after that the full corn in the ear." Mark 4:28. And the prophet Hosea says of Israel, that "he shall grow as the lily." "They shall revive as the corn, and grow as the vine." Hosea 14:5, 7. And Jesus bids us "consider the lilies how they grow." Luke 12:27. The plants and flowers grow not by their own care or anxiety or effort, but by receiving that which God has

HW03/quotes/Bible.htm

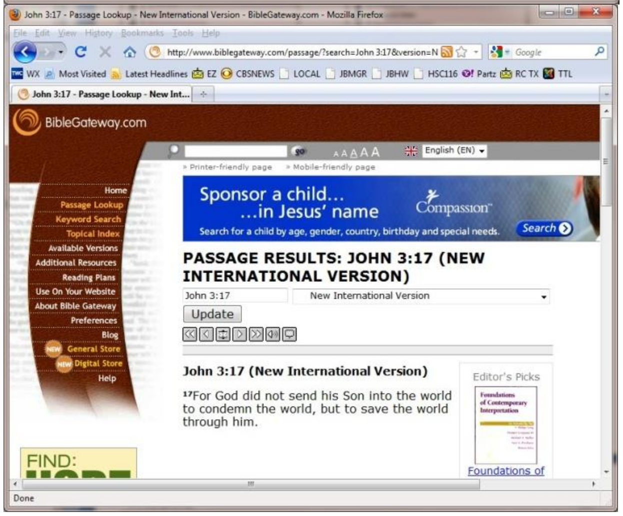

HW03 - Bible Passages - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://hw.cs.southern.edu/jbeckett/HW03/quotes/Bible.htm
Google
WX Most Visited Latest Headlines EZ CBSNEWS LOCAL JBMGR JBHW HSC116 Partz RC TX TTL
HW03 - Bible Passages
W3C HTML 4.0
Done

Bible Passages

Home| Hobby| Quotes| Steps to Christ| Holy Bible

- KJV NIV Psalm 23
- KJV NIV John 3:17

Page at BibleGateway.Com


John 3:17 - Passage Lookup - New International Version - BibleGateway.com - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://www.biblegateway.com/passage/?search=John 3:17&version=N
Google
WX Most Visited Latest Headlines EZ CBSNEWS LOCAL JBMGR JBHW HSC116 Partz RC TX TTL
John 3:17 - Passage Lookup - New Int...
BibleGateway.com
go
A A A A A
English (EN)
» Printer-friendly page
» Mobile-friendly page
Home
Passage Lookup
Keyword Search
Topical Index
Available Versions
Additional Resources
Reading Plans
Use On Your Website
About Bible Gateway
Preferences
Blog
NEW General Store
NEW Digital Store
Help
Sponsor a child…
…in Jesus' name
Compassion
Search for a child by age, gender, country, birthday and special needs.
Search
Update
Editor's Picks
Foundations of Contemporary Interpretation
Foundations of
FIND:
Done

PASSAGE RESULTS: JOHN 3:17 (NEW INTERNATIONAL VERSION)

John 3:17 | New International Version

John 3:17 (New International Version)

[17]For God did not send his Son into the world to condemn the world, but to save the world through him.

Lesson 4 – Images

Before 1992 the Internet was mostly text-only. Even viewing pictures required knowledge of MIME and other then-obscure technologies. The Mosaic browser changed all that by allowing pictures to show in documents. It's done with the `<img>` tag. The picture is not usually contained within the HTML document itself, but is referenced by the document. The browser sees an `<img>`tag and asks for the picture to be sent. This method was originally set up because the Internet had very little bandwidth compared to today, and it was expected that many people would simply ask for pictures to be omitted.

Types of Images

Type	Extension	Compressed?	Transparency	Multiple Frames (Animation)
Bitmap	.bmp	No	No	No
Joint Pictures Expert Group	.jpg .jpeg	Yes (lossy)	No	No
Graphics Interchange Format	.gif	No	Yes	Yes
Portable Network Graphics	.png	Yes (lossless)	Yes	No

Compression

Image files can be large if there is a lot of detail. For instance, a 10-megapixel file could easily consume over 20 megabytes. That's a lot of disk space for just a picture. More importantly for us in this course, that's a terrible amount of bandwidth consumed on the Internet for one picture – which translates to slow page loading. Mathematicians have come to the rescue, with algorithms (methods) that can be used to convey the same information. Some of these algorithms are actually quite simple – such as "This is what blue sky looks like. The top half of the picture is blue sky." Some of them are vastly more complex.

Lossless compression (used in .png)preserves every detail of the original picture. Using the example above, only parts of the sky that are exactly the same color would be referred to as "blue sky." If the sky has various colors of blue, each of them will be treated as a separate zone and will require descriptive information. Note that .png uses the same compression algorithm as .zip files – meaning that you aren't going to save any space by zipping up a .png!

Lossy compression (used in .jpg) goes a step further. The creators of lossy algorithms look at how human vision works, and don't bother preserving things they don't think we'll notice. For instance, most of us don't pay attention to the fact that a blue sky has gradations of color. So we could send only one description of sky color and say the whole sky is that color. Lossy compression algorithms can compress far more than lossless compression algorithms. When you save a .jpg, you can set the quality to various levels. A .jpg file is the most compact, although it may not look as good due to the lossy compression. Nearly-horizontal lines (like roofs on houses) can be a problem for .jpg files.

One final note about compression algorithms: They are sensitive to the amount of detail in a picture. Try it with a camera sometime: Take a picture of a solid color (the sky will do), then take a picture of something with detail like a house. Here are some results from two pictures taken with my Canon TX1:

Picture	Megabytes
Original picture (converted to .bmp to avoid compression)	20.20
.jpg, blue sky with a small branch, quality = 90	1.15
.jpg, neighbor's house, quality = 90	2.96
.jpg, neighbor's house, quality = 50	0.93

Transparency

If an image has transparency, it is possible for backgrounds to show through it. This is something you might like for a logo. An image editor can be used to select what part of the image is transparent, and the percentage of the background which will show through.

bmp

The original picture type, this was adapted from the technology used for television. You will see these only rarely on the Web.

.jpg

The most common picture type on the Web, these are produced by most digital cameras. The compression algorithm can be adjusted for quality when the file is created, which can make for very rapid loading with reasonable quality if certain patterns (such as roof lines) aren't in the picture.

.gif

With transparency and animation supported, the .gif is a great idea except for one problem: To use it legally, you must have a license from CompuServe.

.png

Open-source advocates created this format to replace the legally messy .gif, but they left out animation and extensions permitting animation aren't well supported by browsers. The lossless algorithm used for compression doesn't compress as tightly as .jpg (because it is lossless), but quality is not affected.

.raw

I'm mentioning this only because you might run into this format sometime. This is an uncompressed image taken directory from the sensor of the camera. It has every nuance of detail originally captured. A .raw image is large – like a .bmp. .raw is never used on the Web unless the owners of the Website wish to impress reporters by giving them images suitable for publication.

Using Images

Simple Linking

The simplest way to include a picture on your page an image is to use an anchor tag to link to it as if it were a separate page. Like this:

```
<a href=”mypicture.jpg”>Click to see the pretty picture</a>
```

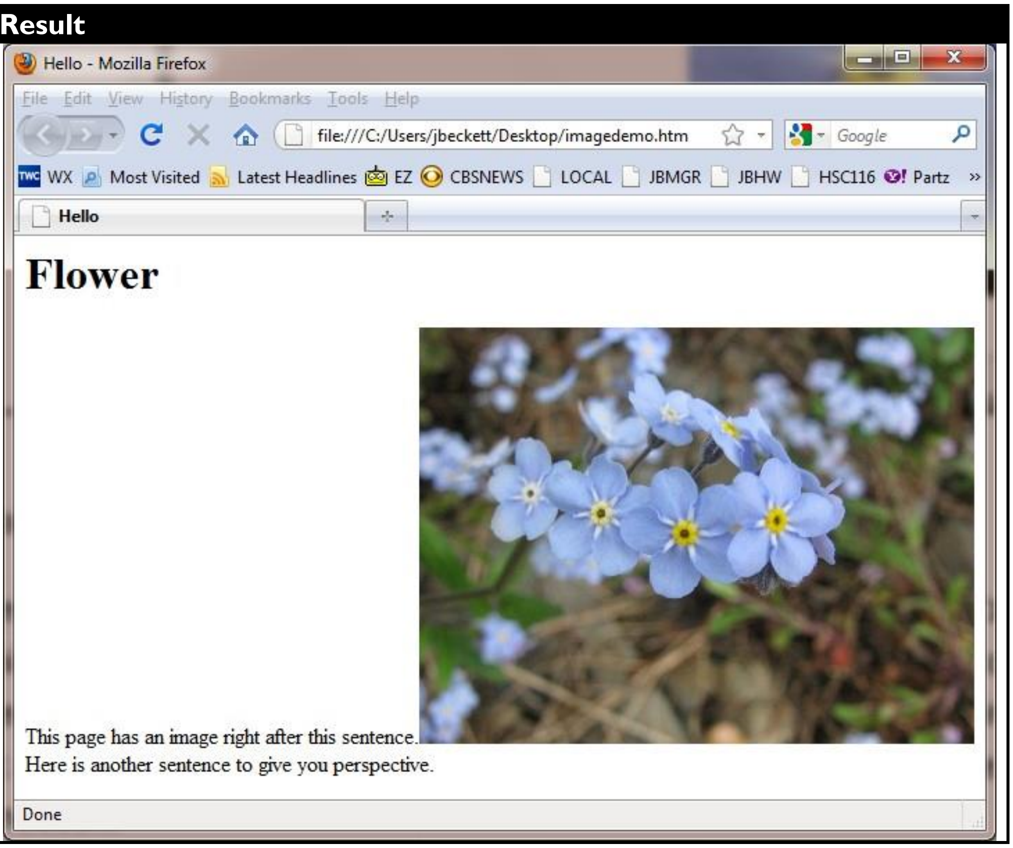
The advantage of this is that the image shows in all its glory in the browser window. The downside of this method is that you can’t see the image and any text at the same time. Which is why this method is not often used alone.

The picture I’m using here is one I took myself – a good way to avoid copyright issues. It was a flower by the side of the path at Icy Straight Point in Alaska. Maybe somebody who is into flowers will tell me what kind it is.

The `<img>` tag

To include an image on a page without any options is simple. Like this:

Code	Result

`<html><head>`
`<title>Hello</title>`
`</head><body>`
`<h1>Flower</h1>`
`<p>This page has an`
`image right after this`
`sentence.`
`<img src="flowers2.jpg"`
`/>`
`Here is another`
`sentence to give you`
`perspective.</p>`
`</body>`
	
	 Hello - Mozilla Firefox
	File Edit View History Bookmarks Tools Help
	file:///C:/Users/jbeckett/Desktop/imagedemo.htm
	Google
	WX Most Visited Latest Headlines EZ CBSNEWS LOCAL JBMGR JBHW HSC116 Partz
	Hello
	Done
	Flower
	This page has an image right after this sentence.
	Here is another sentence to give you perspective.

This simple page demonstrates some challenges that arise when we mix text and pictures:

1. Layout. The browser treats that image as if it were a character like “A” through “z” – but it has to do strange things to get everything on the page because the picture is much taller than the text.
2. Interaction of layout and bandwidth. You can’t see it in a static book, but this page will jump around a bit as it loads on a slow connection.

3. Size. The Web is not a coffee-table magazine; it has far less space to communicate its message. This means that images need to be more focused on the subject and less on the context. (I know, this says a lot about the style of thinking the Web feeds – but that subject is for another class!)

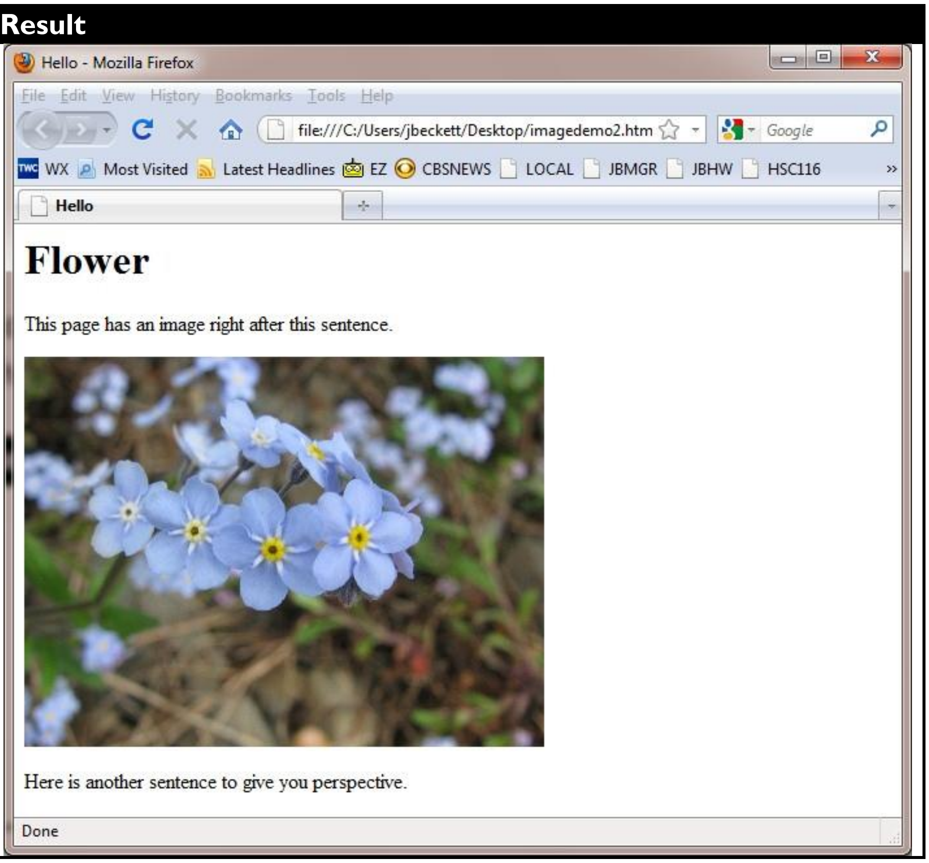
Layout

This is a huge issue, one we'll touch on somewhat briefly in lessons to come. In this lesson we'll use a very simple technique: Not mixing text and graphics. We'll put text ahead of the picture in one paragraph, the picture in its own paragraph, and the text beyond the picture in yet another paragraph. Here's how:

Code

```
<html><head>
<title>Hello</title></head>
<body>
<h1>Flower</h1>
<p>This page has an image
right after this
sentence.</p>
<p><img src="flowers2.jpg"
/>
</p>
<p>Here is another sentence
to give you perspective.</p>
</body>
```

Result


Hello - Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///C:/Users/jbeckett/Desktop/imagedemo2.htm
Google
WX Most Visited Latest Headlines EZ CBSNEWS LOCAL JBMGR JBHW HSC116
Hello
Done

Flower

This page has an image right after this sentence.

Here is another sentence to give you perspective.

This looks better because your site visitor finds things where she might expect them.

Interaction of Layout and Bandwidth

If your site visitor has a slow connection, the page will jump around as it loads. We thought this problem had been solved with hi-speed Internet, but people are forever extending the Internet past its boundaries into new places like cell phones and satellite connection. And these are usually slow. Yes, you can get WiFi on the bus in Chattanooga – but you can bet it isn't like you get on a cable modem at home!

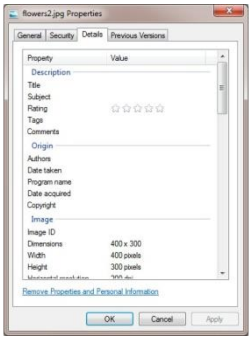
Here's why the bandwidth problem impacts your page if it has pictures. It has to do with the sequence of events when a page with pictures is loaded.

1. Your HTML is downloaded. Since it has no information about how much space to reserve for pictures, none is reserved. So the text appears all together.

2. Your browser sees an "<img>" tag in the HTML, and sends a secondary request to the server asking for that picture. The server responds by sending the picture.
3. When the picture starts loading the browser sees how large it is and re-displays the page with space for the picture.

This problem makes things jump around, which is very frustrating to your site visitor. If your page includes multiple pictures, this re-display operation may happen several times.

The cure for this is to include the specifications of the picture in your HTML. How do you obtain these specs? On Windows, just hover your cursor over the picture icon and you'll see them. Or right-click the picture's icon and select properties, then the Details tab. This picture is 400 pixels high by 300 pixels tall. Here's how the code looks after we implement this:



Property	Value
Description	
Title	
Subject	
Rating	☆☆☆☆☆
Tags	
Comments	
Origin	
Authors	
Date taken	
Program name	
Date acquired	
Copyright	
Image	
Image ID	
Dimensions	400 x 300
Width	400 pixels
Height	300 pixels

Code

```
<html><head><title>Hello</title></head><body>
<h1>Flower</h1>
<p>This page has an image right after this sentence.</p>
<p><img src="flowers2.jpg" height="300" width="400" /></p>
<p>Here is another sentence to give you perspective.</p>
</body>
```

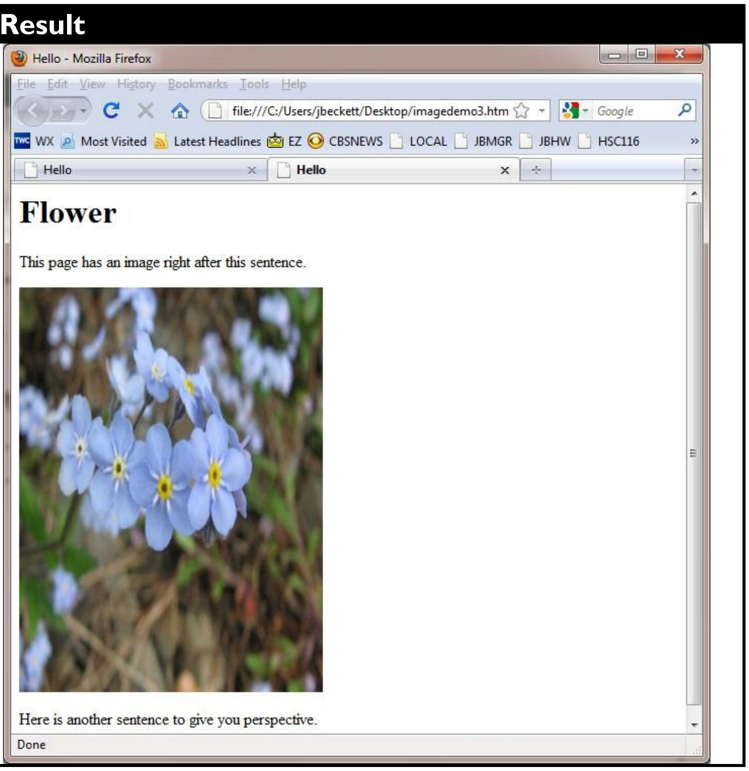
The result is unchanged, but when loaded on a slow connection the image builds smoothly without any jumps or jerks.

It is very important to get these numbers right. Merely switching them does something nasty:

Code

```
<html><head><title>Hello</title></head
><body>
<h1>Flower</h1>
<p>This page has an image right after
this sentence.</p>
<p><img src="flowers2.jpg"
height="400" width="300" /></p>
<p>Here is another sentence to give
you perspective.</p>
</body>
```

Result



Flower

This page has an image right after this sentence.

Here is another sentence to give you perspective.

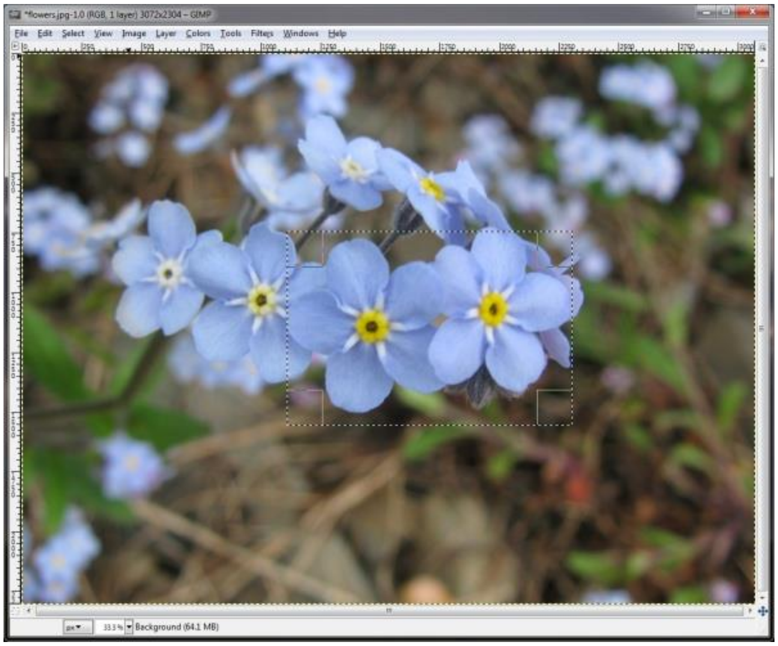
The picture is distorted because the aspect ratio is not the same as when the image was originally made. We're seeing way too much of this on TV these days as we muddle our way toward hi-def – fat people looking skinny and skinny people looking fat. I even see this error on high-profile Websites. It's a quick way to reduce the credibility of your Website.

Another thing you can do (and is an easy way out for people who don't know how to edit images), is to place a very large picture on the server but specify reasonable height and width numbers in the HTML. If there is plenty of bandwidth available nobody notices this. But if bandwidth is a challenge, the page loads much slower. There is an additional disadvantage: You may inadvertently include images of much higher quality than you intended. If your family pictures are placed in full resolution on your Website, they can be captured by ad agencies to and used for promoting products you didn't even know existed. Illegal, probably. But do you know a lawyer who can file your case in Slovenia?

Aspect Ratio is the relationship between an images width and its height. 4:3 is the traditional ratio used in television. The most common "widescreen" aspect ratio is 16:9, although many others are used. DSLR cameras may use a different aspect ratio such as 2:3.

Size

This picture shows a lot of weeds compared to the flowers. What to do? Take out the weeds! Doing this requires us to use an image editing program. If you have PhotoShop and know how to use it, crop and resample. Don't have PhotoShop? Here's how to do it with Gimp 2.6:


*flowers.jpg-1.0 (RGB, 1 layer) 3072x2304 – GIMP
File Edit Select View Image Layer Colors Tools Filters Windows Help
px
33.3%
Background (64.1 MB)

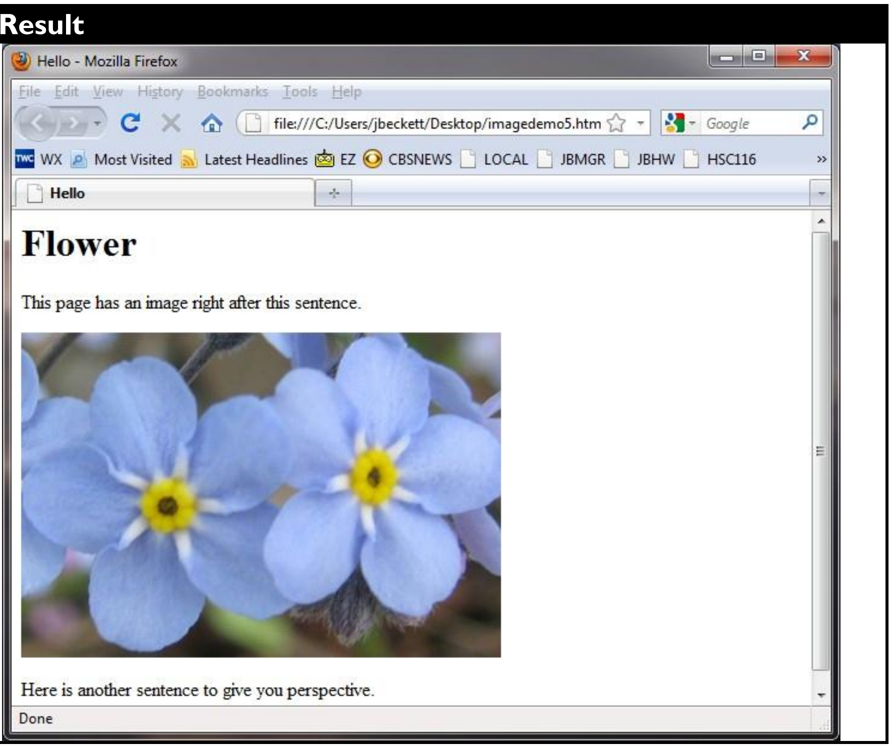
1. Open the picture in Gimp with File..Open.
2. Gimp offers to rotate the picture. Decline by clicking "Keep Orientation" unless it looks wrong.
3. In the Toolbox, click the "Rectangle Select Tool" .
4. Click one corner of the area you want to preserve, drag to the opposite corner, and let go. It looks like the picture on the right. Note the box with small boxes, neatly framing the two flowers I wish to feature.
5. Click Image..Crop To Selection. The rest of the picture will disappear.
6. Click Image..Scale Image. The "Scale Image" applet will come up. Click in the Width box and enter the value you'd like (I selected 400 for this example). Press tab and it will automatically adjust the Height to preserve the aspect ratio. Click Scale
7. Click File..Export and save your new file.

The page now looks like this – undistorted, and loads very fast:

Code

```
<html><head><title>Hello</title></head><body>
<h1>Flower</h1>
<p>This page has an image
right after this sentence.</p>
<p><img src="flowers5.jpg"
height="271" width="400"
/></p>
<p>Here is another sentence to
give you perspective.</p>
</body>
```

Result



Flower

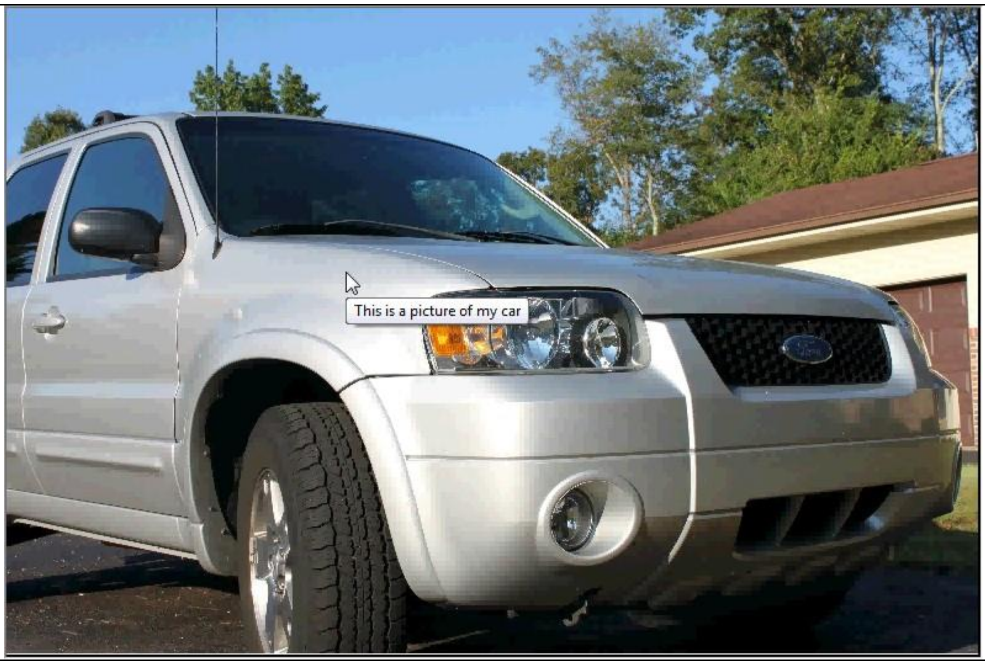
This page has an image right after this sentence.

Here is another sentence to give you perspective.

Image Title

If you include a title in an <img> tag, hovering the mouse over the image will display the title. Like this:

```
<img
src="mycar.jpg"
title="This is a
picture of my car"
/>
```



(If you see this car around time, it isn't mine any more – I gave it to a relative!)

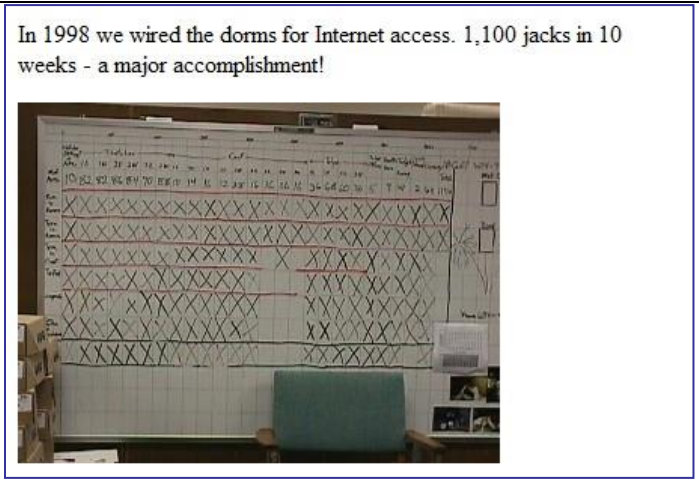
alt=

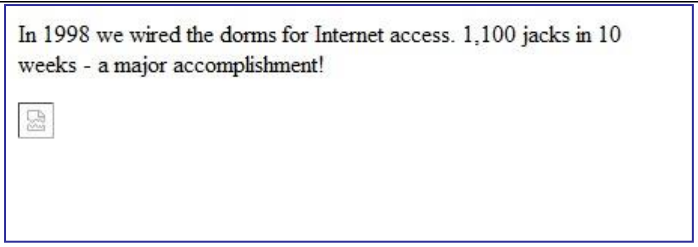
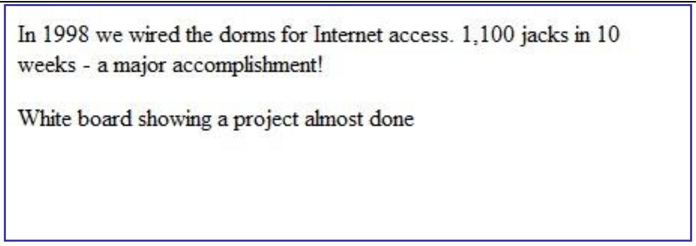
Your page should have alt= attributes for all images. The purpose of this attribute is to provide information about the picture. Why do you need information about a picture, when "a picture is worth 1,000 words?" Because pictures don't necessarily look to your site visitor the way they look to you. Differences include:

1. The visitor's computer or Internet connection may not be as good as yours, so they may have configured their browser not to display images.
2. The visitor may have poor or no vision (blind), perhaps using software which assists them by reading the words on your page aloud.

These visitors need you to explain what the picture means. That's what alt= is for. Use this attribute to explain what is in the image – it's as simple as that.

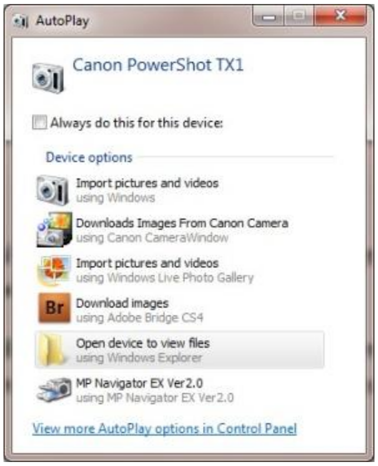
Here is an example.

`<p>In 1998 we wired the dorms for`
`Internet access.  1,100 jacks in 10`
`weeks`
` - a major accomplishment!</p>`
`<img alt="White board showing a`
`project almost done"`
`src="AlmostDone.jpg" />`
 	
HTML Code
	
	In 1998 we wired the dorms for Internet access. 1,100 jacks in 10 weeks - a major accomplishment!
	
	Normal view in a browser

 In 1998 we wired the dorms for Internet access. 1,100 jacks in 10 weeks - a major accomplishment! View in a browser with images disabled, without alt=	 In 1998 we wired the dorms for Internet access. 1,100 jacks in 10 weeks - a major accomplishment! White board showing a project almost done View in a browser with images disabled, with alt=

Loading From a Camera

So, somebody hands you a digital camera so you can use the images from it. How do you get them on your computer? Here are two methods, one of which will probably work.


AutoPlay
Canon PowerShot TX1
☐ Always do this for this device:
Device options
Import pictures and videos
using Windows
Downloads Images From Canon Camera
using Canon CameraWindow
Import pictures and videos
using Windows Live Photo Gallery
Download images
using Adobe Bridge CS4
Open device to view files
using Windows Explorer
MP Navigator EX Ver2.0
using MP Navigator EX Ver2.0
View more AutoPlay options in Control Panel

1. Find a USB cable that will connect the camera to your computer. Most cameras come with such a cable. Plug it in. If nothing happens, you may have to turn the camera on and perhaps set it to "Play" mode. Wait a few moments for a driver to be installed. Once you are told the device is ready to use, wait another moment. Hopefully an "AutoPlay" window like the one on the right should appear.
2. Remove the memory chip from the camera and plug it into a Flash reader or a camera memory slot in your computer. The AutoPlay window should appear.

Now you can select "Open Device to view files" and click your way in to wherever the pictures are. If the AutoPlay window does not appear, click the windows button and select "Computer", then look for your camera and click on it.

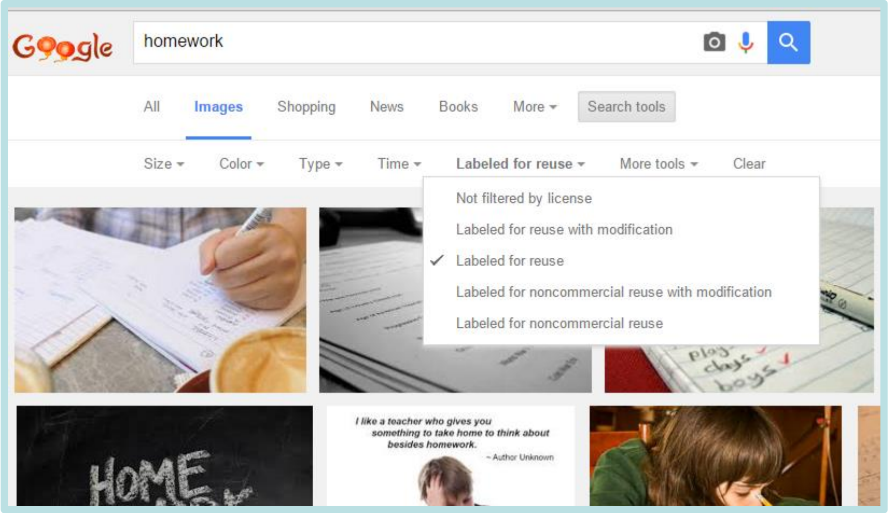
Now you can drag your pictures to a directory on your computer.

Images from the Web

The picture you want to use may very well be out there somewhere on the Web. One good place to look is http://images.google.com. Is there a way you can get it onto your computer? Probably. Here are some methods that may work.

- Right-click the image and select "Save Image As".
- View the source of the page you are looking at, seeing if you can find the <img> tag used for the picture you want. Then splice that into the URL (this can take creativity). This may put the image in your browser window alone, so that you can save it.
- If nothing else works, you can almost always use a screen capture utility. Check the CD that came with your digital camera for a program called Photo Studio – it can do the job.

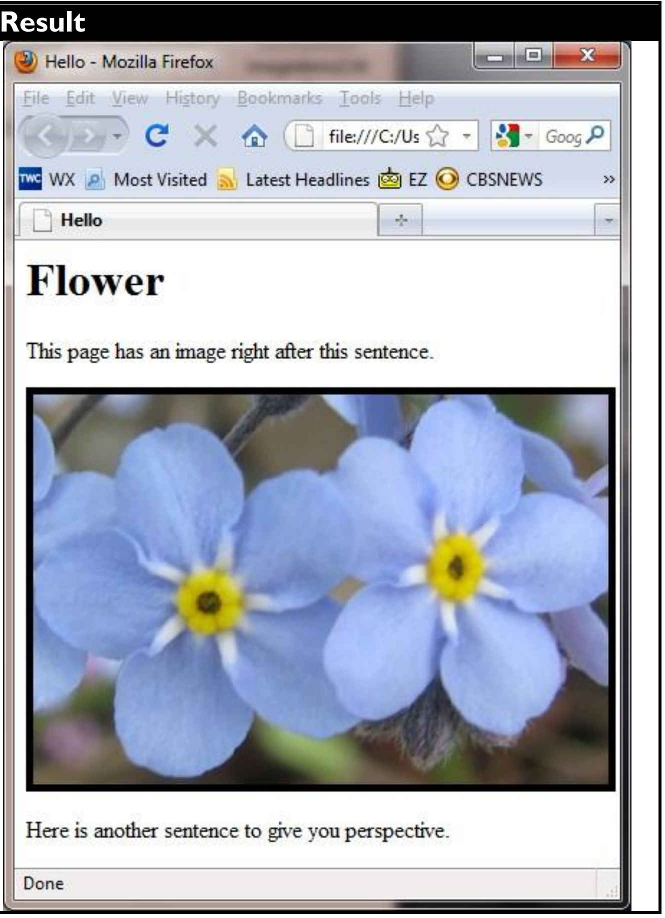
Regardless of how you obtain an image, you are responsible for complying with copyright law. How? My favorite way is to use https://images.google.com. Select "Search tools," "Usage Rights," and "Labeled for reuse." Google will limit your results to items you may use legally in this case.


Google
homework
All
Images
Shopping
News
Books
More
Search tools
Size
Color
Type
Time
Labeled for reuse
More tools
Clear
Not filtered by license
Labeled for reuse with modification
✓ Labeled for reuse
Labeled for noncommercial reuse with modification
Labeled for noncommercial reuse
I like a teacher who gives you something to take home to think about besides homework.
~ Author Unknown
HOME

Borders

An image may have a border. Here's an example showing a 5-pixel border using the "`border=`" attribute:

Code	Result

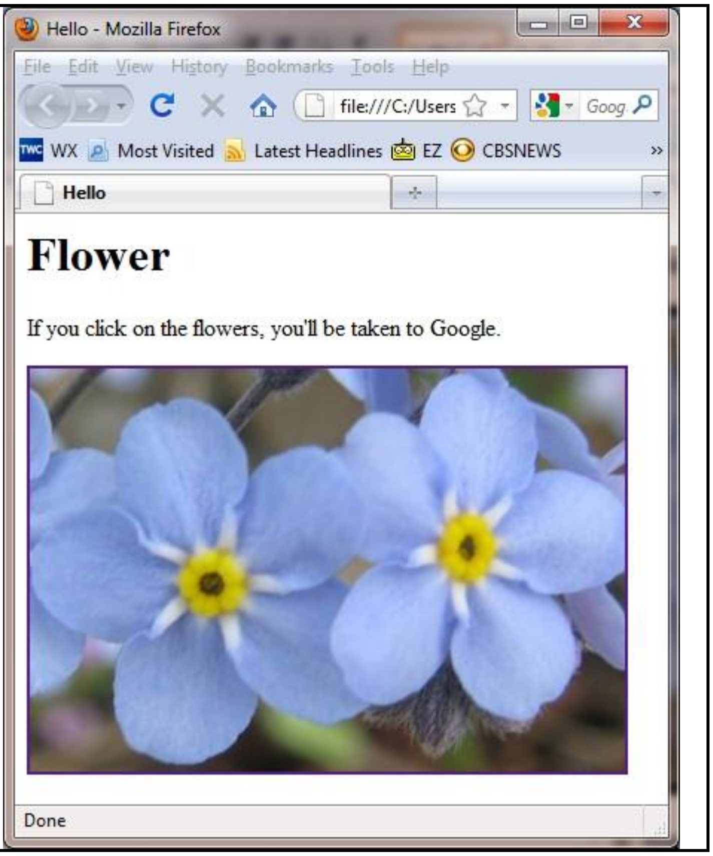
`<html><head><title>Hello</title></head>`
`<body>`
`<h1>Flower</h1>`
`<p>This page has an image right after this sentence.</p>`
`<p>`
`<img src="flowers5.jpg" height="271"`
`  width="400" border="5" />`
`</p>`
`<p>Here is another sentence to give you perspective.</p>`
`</body>`
	
	 Hello - Mozilla Firefox
	File Edit View History Bookmarks Tools Help
	file:///C:/Us
	Goog
	WX Most Visited Latest Headlines EZ CBSNEWS
	Hello
	Done
	Flower
	This page has an image right after this sentence.
	Here is another sentence to give you perspective.

Using an image as a link

To use an image as a button that works as a hyperlink, use the `<img>` tag as the text of the hyperlink. Like this:

Code	Result

```
<html><head>
<title>Hello</title>
</head><body>
<h1>Flower</h1>
<p>If you click on the flowers, you'll
be taken to Google.</p>
<p>
<a href="http://google.com">
<img src="flowers5.jpg" height="271"
width="400"  /></a>
</p>
</body>
```


Hello - Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///C:/Users
Goog
WX Most Visited Latest Headlines EZ CBSNEWS
Hello
Done

Flower

If you click on the flowers, you'll be taken to Google.

Note that doing this puts a blue border around the image, so it will appear to your visitors as a link. Artistically it might be better to delete the border using "border=0", but this is compromises functionality so I don't recommend it.

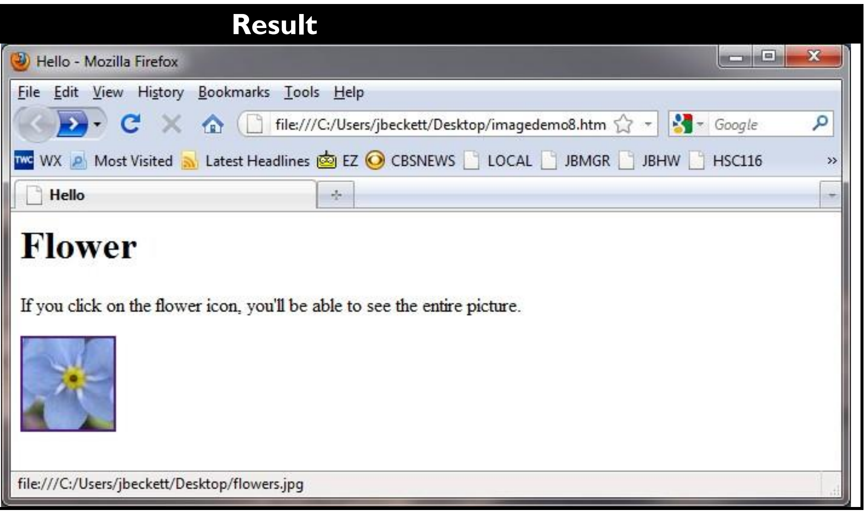
Thumbnails

Putting these techniques together, you can use a “thumbnail” image as a link to the full image. Use an image editor to create a small image, then code the image as a link to the full picture. Like this:

Code

```
<html><head>
<title>Hello</title></head>
<body>
<h1>Flower</h1>
<p>If you click on the flower
icon, you'll be able to see the
entire picture.</p>
<p>
<a href="flowers.jpg">
<img src="FlowerIcon.jpg" />
</a></p>
</body>
```

Result


Hello - Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///C:/Users/jbeckett/Desktop/imagedemo8.htm
Google
WX Most Visited Latest Headlines EZ CBSNEWS LOCAL JBMGR JBHW HSC116
Hello

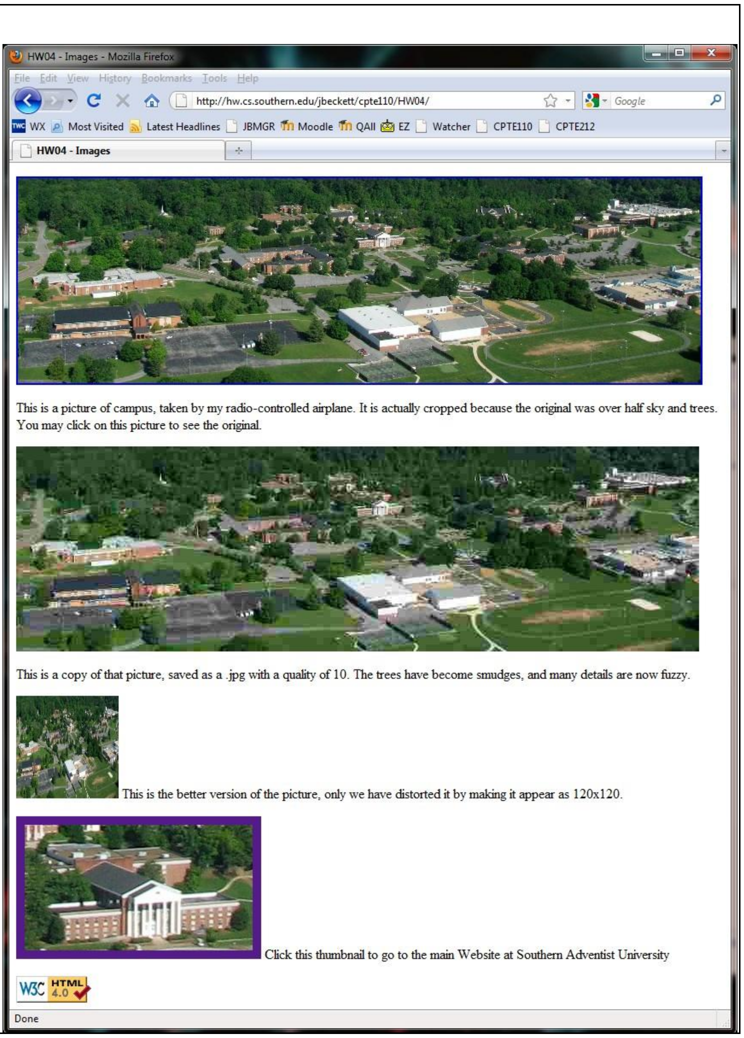
Flower

If you click on the flower icon, you'll be able to see the entire picture.

file:///C:/Users/jbeckett/Desktop/flowers.jpg

HW04: Images

- Obtain campus.jpg from the support Website.
- Crop the picture so it shows only the campus, not blue sky and the hill of trees behind.
- Resample the cropped picture so that it is 600 pixels wide. Save it in .jpg format with high (probably 90) quality.
- Save another copy of the processed picture from the previous step, with a quality of 10.
- Go back to the original campus.jpg, and create a cropped image showing only Wright Hall.
- Create a page like the one on the right. It must do the things mentioned in the descriptions. The border on the last picture should be 10 pixels wide.
- Name your page “index.html” and place it in your HW04 directory, so it will come right up when someone clicks on HW04.


HW04 - Images - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://hw.cs.southern.edu/jbeckett/cpte110/HW04/
Google
WX Most Visited Latest Headlines JBMGR Moodle QAII EZ Watcher CPTE110 CPTE212
HW04 - Images
W3C HTML 4.0
Done

This is a picture of campus, taken by my radio-controlled airplane. It is actually cropped because the original was over half sky and trees. You may click on this picture to see the original.

This is a copy of that picture, saved as a .jpg with a quality of 10. The trees have become smudges, and many details are now fuzzy.

This is the better version of the picture, only we have distorted it by making it appear as 120x120.

Click this thumbnail to go to the main Website at Southern Adventist University

Lesson 5 – Tables

Tables are a powerful feature of HTML. They are useful in several ways:

- Presenting tabular data (duh)
- Structuring a page so things go where the designer wants them to go
- Structuring a page so that variable amounts of information are displayed properly

A table requires the following tags:

Tag or Parameter	Purpose
`<table>`	Define the overall table
`<tr>`	Define a table row
`<td>`	Define a table data element

In addition, we may use the <th> tag explained below and the <thead>, <tbody>, and <tfoot> which are not covered in this book.

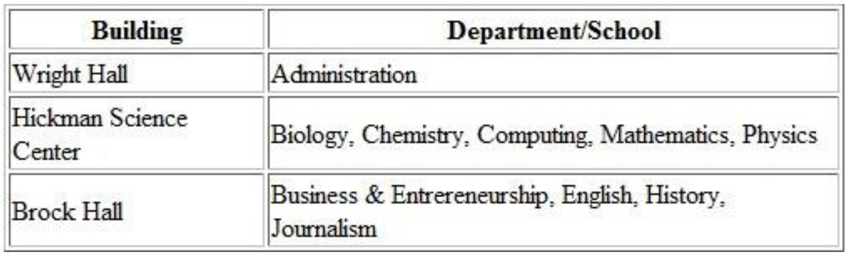
Here's a simple table to begin with. We'll expand it as we go.

Code

```
<table border="1">
  <tr>
    <th>Building</th>

<th>Department/School</th>
  </tr>
  <tr>
    <td>Wright Hall</td>
      <td>Administration</td>
  </tr>
  <tr>
      <td>Hickman Science
Center</td>
      <td>Biology, Chemistry,
Computing, Mathematics,
Physics</td>
  </tr>
  <tr>
    <td>Brock Hall</td>
    <td>Business &
Entrereneurship, English,
History, Journalism</td>
  </tr>
</table>
```

Result



Building	Department/School
Wright Hall	Administration
Hickman Science Center	Biology, Chemistry, Computing, Mathematics, Physics
Brock Hall	Business & Entrereneurship, English, History, Journalism

The border attribute of the `<table>` tag is zero by default, meaning that the table would have no border. Often you will want this, but I recommend that when building a table you set it to so you can see what is happening. Once you have your table functioning properly, you might wish to remove the lines by setting the border to 0.

The HTML code for a table is organized in outline form with the rows a major item and the columns a minor item. As we've seen before it doesn't matter how you space the HTML. I've indented mine neatly so you can read it easier.

You are probably acquainted with the concept of a “cell” from using a word processor or spreadsheet program. In HTML there are at least two types of cells:

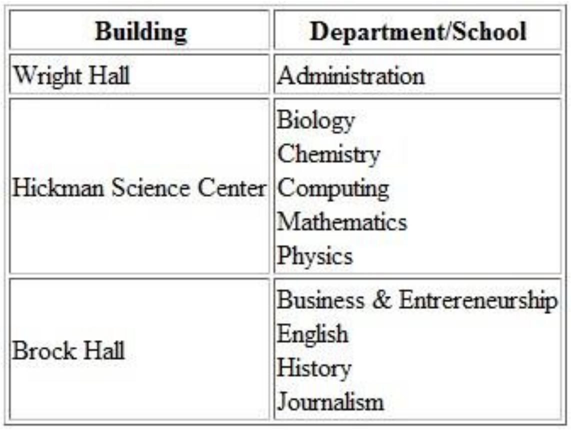
- <th> means “table heading cell”. It is rendered with the text centered and bold-faced.
- <td> means “table data cell”. It is rendered with the text flush left.

In the example above, each row is framed with a <tr> tag. Each item within a row is framed with either <th> or <td>. <th> causes the item to be centered and bold-faced. <td> items are flush-left and not bold-faced unless you add a <b> tag.

The table would look a little better if we put the department names on different lines. For this we’ll start by using the <br /> tag, which simply inserts a line break.

Code **Result**

```
<table border="1">
  <tr>
    <th>Building</th>
    <th>Department/School</th>
  </tr>
  <tr>
    <td>Wright Hall</td>
     <td>Administration</td>
  </tr>
  <tr>
     <td>Hickman Science Center</td>
     <td>Biology<br />Chemistry<br
/>Computing<br />Mathematics<br
/>Physics</td>
  </tr>
  <tr>
    <td>Brock Hall</td>
    <td>Business & Entrereneurship<br
/>English<br />History<br
/>Journalism</td>
  </tr>
</table>
```

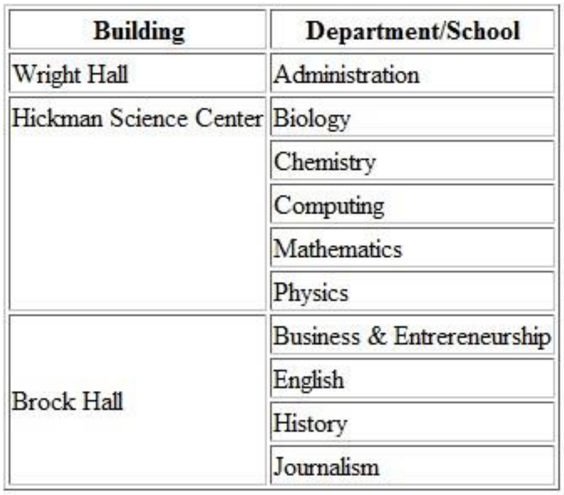


Building	Department/School
Wright Hall	Administration
Hickman Science Center	Biology Chemistry Computing Mathematics Physics
Brock Hall	Business & Entrereneurship English History Journalism

But what if we wish to place lines between the departments on the right while keeping the building name blocks on the left so they cover multiple departments? We can do that. It will require the `<rowspan>` tag.

Code **Result**

```
<table border="1">
  <tr>
    <th>Building</th>
    <th>Department/School</th>
  </tr>
  <tr>
    <td>Wright Hall</td>
      <td>Administration</td>
  </tr>
  <tr valign="top">
      <td rowspan="5">Hickman Science
Center</td>
      <td>Biology</td>
  </tr>
    <tr><td>Chemistry  </td></tr>
    <tr><td>Computing  </td></tr>
    <tr><td>Mathematics</td></tr>
    <tr><td>Physics    </td></tr>
  <tr>
    <td rowspan="4">Brock Hall</td>
    <td>Business & Entrereneurship</td>
  </tr>
    <tr><td>English   </td></tr>
    <tr><td>History   </td></tr>
    <tr><td>Journalism</td></tr>
</table>
```



Building	Department/School
Wright Hall	Administration
Hickman Science Center	Biology
	Chemistry
	Computing
	Mathematics
	Physics
Brock Hall	Business & Entrereneurship
	English
	History
	Journalism

The `<rowspan>` tag indicates that this item should be considered repeated for spacing purposes. So subsequent lines leave space for that item.

I've changed the spacing in my code, placing the additional lines in a group indented below the line that started that group. This has nothing to do with how the result works – only making it more readable when a person is looking at the code.

The difference between the positioning of Hickman Science Center and Brock Hall is deliberate. This is to illustrate that you can select the alignment of text in a block when it has a `rowspan` attribute. `valign` is often used when we lay out a page with a menu, so that the menu stays at the top of a block.

A Larger Example

Here is a diagram showing the organization of the New Testament of the Holy Bible, illustrating how the compilers grouped its books together by authorship as well as chronology. It uses both rowspan and colspan. Again, I've indented the code for clarity – but contracted the indentation for the epistles to save space in this diagram. But how I indented the code has nothing to do with how it is displayed.

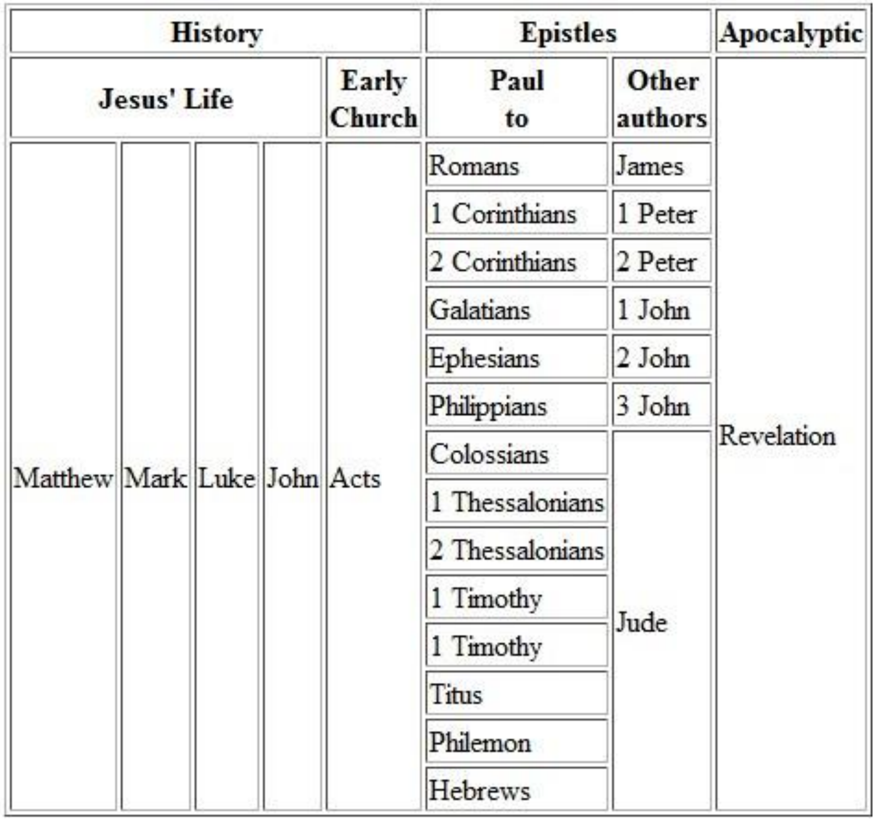
SOUTHERN ADVENTIST UNIVERSITY School of Computing

I make no apology for using a religious text in a technical book. This is an excellent example of information organization that dates back well over 1,000 years. This structure was apparently borrowed from the Hebrew Bible that preceded it by centuries. If an adopter of this book wishes to change this page to another context, they are welcome to search out a better example.
Techies may note that scholars differ on some of the fine points, such as the authorship of the epistle to the Hebrews and the fact that I haven't shown the authorship of Acts. Discuss away – you're talking about Good Stuff.

Code

```
<table border="1">
<tr>
  <th colspan="5">History</th>
  <th colspan="2">Epistles</th>
  <th>Apocalyptic</th>
</tr>
<tr>
  <th colspan="4">Jesus'
Life</th>
  <th>Early<br />Church</th>
  <th>Paul<br />to</th>
  <th>Other<br />authors</th>
  <td
rowspan="15">Revelation</td>
</tr>
<tr><td
rowspan="15">Matthew</td>
    <td rowspan="15">Mark</td>
      <td
rowspan="15">Luke</td>
      <td
rowspan="15">John</td>
      <td
rowspan="15">Acts</td>
      <td>Romans</td>
      <td>James</td>
</tr>
<tr><td>1 Corinthians</td>
    <td>1 Peter</td></tr>
<tr><td>2 Corinthians</td>
    <td>2 Peter</td></tr>
<tr><td>Galatians</td>
    <td>1 John</td></tr>
<tr><td>Ephesians</td>
    <td>2 John</td></tr>
<tr><td>Philippians</td>
    <td>3 John</td></tr>
<tr><td>Colossians</td>
    <td
rowspan="8">Jude</td></tr>
<tr><td>1
Thessalonians</td></tr>
<tr><td>2
Thessalonians</td></tr>
<tr><td>1 Timothy</td></tr>
<tr><td>1 Timothy</td></tr>
<tr><td>Titus</td></tr>
<tr><td>Philemon</td></tr>
```

Result



History					Epistles		Apocalyptic
Jesus' Life				Early Church	Paul to	Other authors	Revelation
Matthew	Mark	Luke	John	Acts	Romans	James	
					1 Corinthians	1 Peter	
					2 Corinthians	2 Peter	
					Galatians	1 John	
					Ephesians	2 John	
					Philippians	3 John	
					Colossians	Jude	
					1 Thessalonians		
					2 Thessalonians		
					1 Timothy		
					1 Timothy		
					Titus		
					Philemon		
					Hebrews		

```
<tr><td>Hebrews</td></tr>
</table>
```

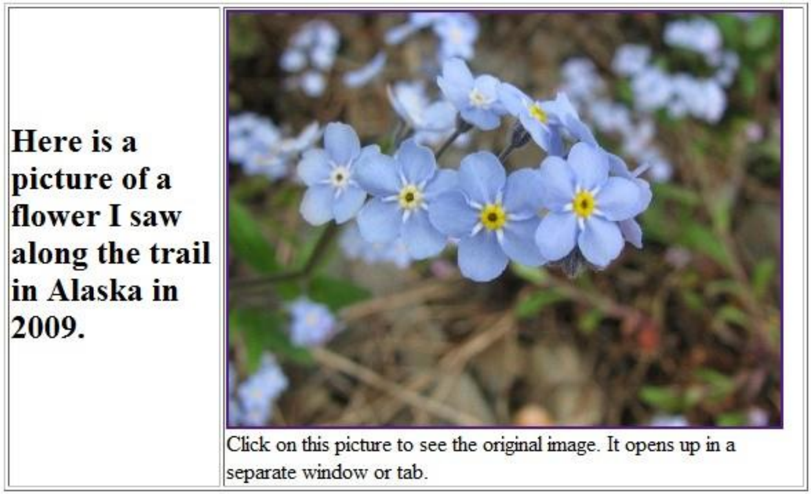
Tables and pictures

Tables are a great way to organize pages that include pictures. Using a table avoids the alignment issues that arise when we try to combine text and pictures. Let's re-visit that flower page from Lesson 4:

Code

```
<table border="1">
<tr>
  <td width="150"><h2>Here is a
picture of a flower I saw along
the trail in Alaska in
2009.</h2></td>
  <td><a href="flowers.jpg"
target="_blank"><img
src="flowers2.jpg" /></a><br />
      Click on this picture to
see the original image.  It opens
up in a separate window or
tab.</td>
</tr>
</table>
```

Result



Here is a picture of a flower I saw along the trail in Alaska in 2009.

Click on this picture to see the original image. It opens up in a separate window or tab.

To include a picture in a table data element, we simply placed the <img> tag where we want it. We positioned text under the picture by using a <br /> tag to put them on separate lines.

I was unhappy with the size of font on the left side, so I used <h2> to make it bigger.

alt=

This attribute of an <img> tag is used by handicap software to help interpret your page for blind people (yes, they're on the Web). Blind-reading software works well with tables, especially if you use <th> for headings.

title=

This attribute of an <a> tag causes a pop-up to appear if the user hovers over a link. This gives the site visitor warning about what might happen if they click on the link.

Do you see how the various tools in HTML sort of snap together like Lego™ blocks? While at first it may seem confusing, it's relatively simple: You build complex things by combining simple elements. One thing you must remember, however, is the rule that however you get into something, you back out of it in reverse order. So it would be improper to code:

```
<b><i>something</b></i>
```

The proper way would be:

```
<b><i>something</i></b>
```

Did you skip reading that example? Shame on you! This is one of the most important concepts in the entire book. Now go back, and pay attention this time. I promise you'll get your effort back when you are doing the assignment for this lesson.

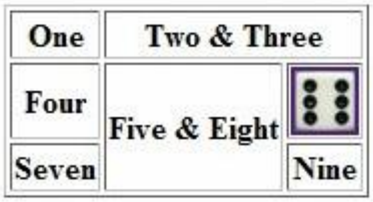
Putting It Together

Links, images, and tables work together to make pages that are highly functional. Consider this page:

Code

```
<table border="1">
  <tr>
    <th>One</th>
    <th colspan="2">Two & Three</th>
  </tr>
  <tr>
    <th>Four</th>
    <th rowspan="2">Five & Eight</th>
    <th>
        <a href=http://abclocal.go.com/wpvi/index
          alt=”WPVI logo” title=”WPVI” >
          <img src="images/dice6.jpg" />
        </a>
    </th>
  </tr>
  <tr>
    <th>Seven</th>
    <th>Nine</th>
  </tr>
</table>
```

Result



One	Two & Three	
Four	Five & Eight	⚅
Seven		Nine

This table is basically 9 columns by 9 rows. We’ve used “<th>” instead of “<td>” to make everything centered, and we’ve omitted any width numbers in our cells so that the browser would automatically adjust as needed.

We have used “colspan=” to make the top row consist of only two cells, with the right cell spanning the second and third columns. And we have used “rowspan= to make the middle cell extend down to cover the bottom row as well.

What do we do with the extra cells? We simply omit them! The browser knows they aren’t needed.

How do you center everything within a table cell? Easy: <td align="center">

Developing a Table with "rowspan=" and/or "colspan="

Try the following method:

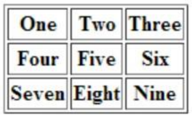
1. Lay out your code as if there were no spans involved. Use only text – no pictures or links.
2. Use indentation freely to make things easier to read. No need to make it hard on yourself!
3. Check to see that everything is in place.
4. One by one, add the spans. Each time, delete the cell that will no longer exist.
5. Now add in your pictures and make them links.

Use indentation freely to make things easier to read. No need to make it hard on yourself! The browser doesn't care, and it doesn't really slow down your pages.

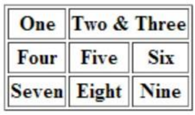
Here's the "before" and "after" as we put in that colspan.

Before	Planning Changes	After

<table border="1">
  <tr>
    <th>One</th>
    <th>Two</th>
    <th>Three</th>
  </tr>
    <tr>
    <th>Four</th>
    <th>Five</th>
    <th>Six</th>
  </tr>
  <tr>
    <th>Seven</th>
    <th>Eight</th>
    <th>Nine</th>
  </tr>
</table>
	<table border="1">
	  <tr>
	    <th>One</th>
	    <th colspan="2">
	      Two & Three
	    </th>
	    ~~<th>Three</th>~~
	  </tr>
	    <tr>
	    <th>Four</th>
	    <th>Five</th>
	    <th>Six</th>
	  </tr>
	  <tr>
	    <th>Seven</th>
	    <th>Eight</th>
	    <th>Nine</th>
	  </tr>
	</table>
		<table border="1">
		  <tr>
		    <th>One</th>
		    <th colspan="2">
		      Two & Three
		    </th>
		  </tr>
		    <tr>
		    <th>Four</th>
		    <th>Five</th>
		    <th>Six</th>
		  </tr>
		  <tr>
		    <th>Seven</th>
		    <th>Eight</th>
		    <th>Nine</th>
		  </tr>
		</table>



One	Two	Three
Four	Five	Six
Seven	Eight	Nine



One	Two & Three	
Four	Five	Six
Seven	Eight	Nine

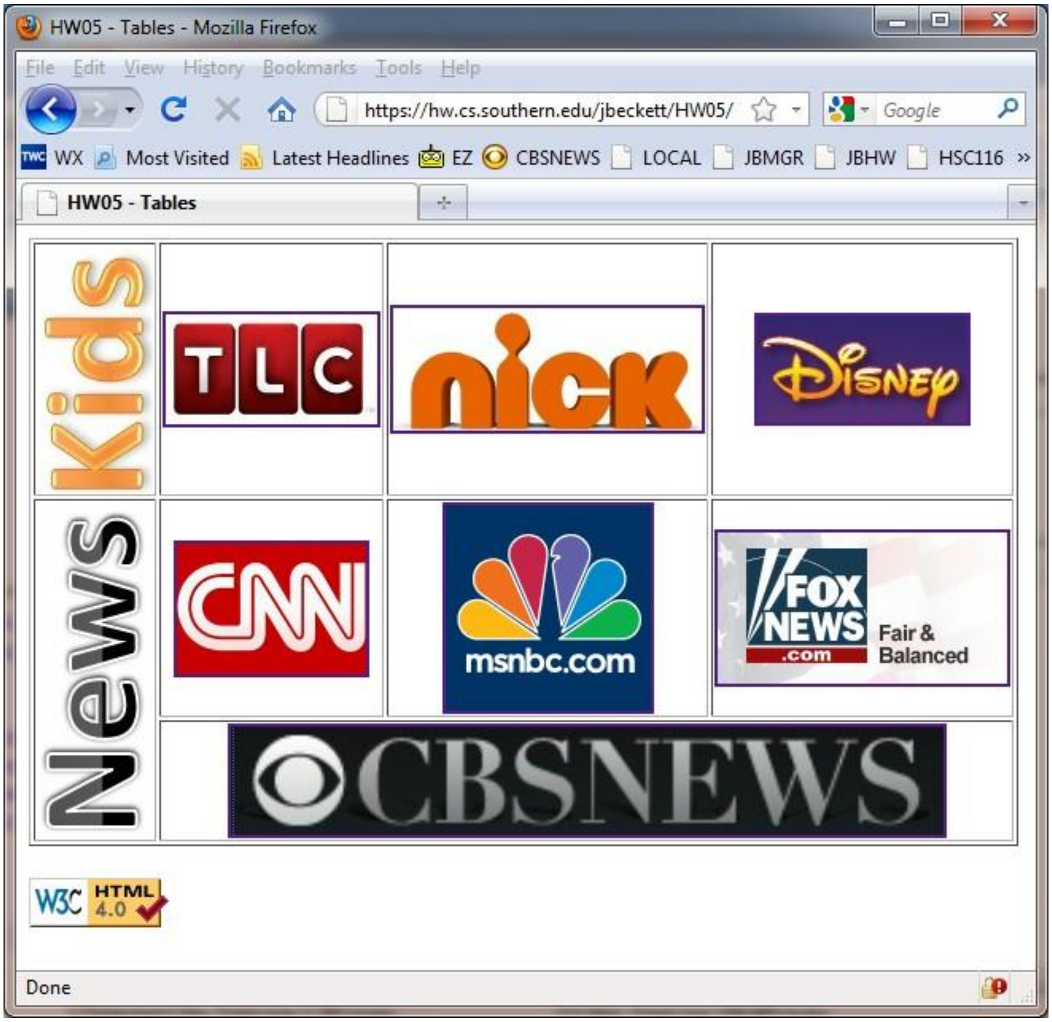
Changing "Six" into an image is easy, but it takes three steps

Original Cell	<th>Six</th>
Add a Link	<th> <a href="http://abclocal.go.com/wpvi/index">Six</a> </th>

Use Picture	<th>
	  <a href="http://abclocal.go.com/wpvi/index">
	    <img src="images/dice6.jpg" />
	  </a>
	</th>

Put in the trimmings	<th>
	  <a href=http://abclocal.go.com/wpvi/index
	     alt=”WPVI logo” title=”WPVI” >
	    <img src="images/dice6.jpg" />
	  </a>
	</th>

HW05: This assignment combines links and images in a table. Create a page that displays the network logos as shown using a table. Set up each logo image so that it has an appropriate alt= attribute, forms a link to the network cited, and uses title= to display the network name when the mouse hovers over it. See the support Website for the logo images.


HW05 - Tables - Mozilla Firefox
File Edit View History Bookmarks Tools Help
https://hw.cs.southern.edu/jbeckett/HW05/
Google
WX Most Visited Latest Headlines EZ CBSNEWS LOCAL JBMGR JBHW HSC116 »
HW05 - Tables
Kids
TLC
nick
Disney
News
CNN
msnbc.com
FOX NEWS .com Fair & Balanced
CBSNEWS
W3C HTML 4.0
Done

Links to use:

Network	URL
The Learning Channel	http://tlc.discovery.com
Nickelodeon	http://www.nick.com
Disney	http://home.disney.com
Cable News Network	http://www.cnn.com
MSNBC	http://msnbc.com
Fox News	http://foxnews.com
CBS News	http://cbsnews.com

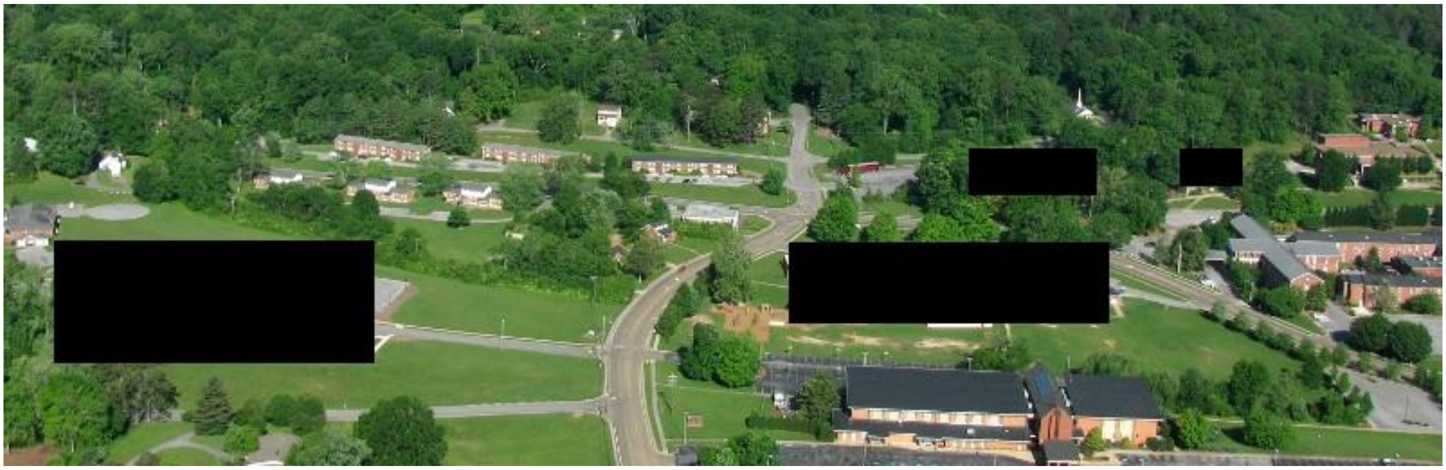
Lesson 6 – Image Maps, Audio/Video

This lesson is about dynamic behavior of a Web page, which means there is no way to illustrate the results in a book. The support site has a demo that shows all the examples in this lesson – except for playing an audio without anything showing on the page.

Image Maps

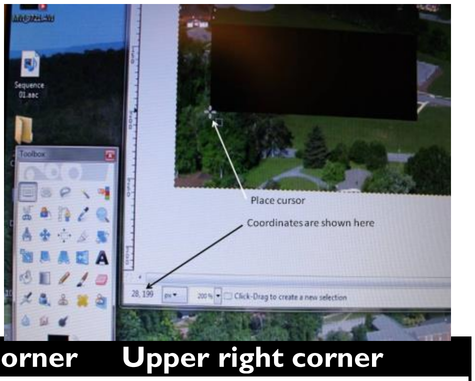
An image map is a great tool for communicating with your users. It allows them to make their click choices based on areas of a picture rather than text. Here's an example: an aerial picture of part of my campus.

I would like to put this picture on a Website, and set it up so that people can click on buildings to activate various functions of the site. The first step is to establish "hot zones" – places where clicking will do something. Here's another copy of the picture with the hot zones I've selected marked in black. Note that this is a copy I made of the image, one that will not actually be posted on the Website.



To identify those hot zones, I need to find their numerical coordinates. We'll need coordinates for the two opposite corners of each rectangular zone. Gimp lets us do this. If we pull up the picture in Gimp and place our cursor at the location of interest, it will show us the coordinates as shown on the right.

Here are the coordinates for several buildings I'd like to use in my site map:


Sequence 01.aac
Toolbox
Place cursor
Coordinates are shown here
28, 199
px
200 %
Click-Drag to create a new selection

Building or Group	Department	Lower left corner	Upper right corner
Southern Village	Advanced student housing	28,199	205,130
Spalding School	K-8 school	436,177	613,132
Hickman Science Center	Bio, Chem, CS, Math, Physics	535,105	606,80
Herin Hall	Education & Psych.	653,100	687,80

Now to create the HTML. Here's the code used for the demo on our support Website:

Code

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Map Assignment</title>
</head>

<body>
<p> Click a spot on the map below to view information about our buildings.<p>
<map name="campusmap">
<area target="_blank" href="sovill.htm"   shape="rect" coords=" 28,199,205,130" title="Southern Village" />
<area target="_blank" href="spalding.htm" shape="rect" coords="436,177,613,132" title="Spalding School" />
<area target="_blank" href="hickman.htm"  shape="rect" coords="525,105,606, 80" title="Hickman Science
Center" />
<area target="_blank" href="herin.htm"    shape="rect" coords="653,100,687, 80" title="Herin Hall" />
</map>
<img border="0" src="southendsmall.jpg" usemap="#campusmap" align="left" />
</body>
</html>
```

Tags Used for Image Maps

A map involves three different tags:

1. <map> - in which you give this map a name. This name doesn't appear on your Website. It is merely a designation you'll use in your code.
2. Within the <map> tag appear <area> tags describing each of the hot zones. These have a number of attributes:
   - target – where you want the new URL to appear. I used "_blank" because I wanted a separate window or tab. Omit target to use the same window.
   - href – the document you want to display if this hot zone is clicked.
   - shape – we're specifying a rectangle.
   - cords – coordinates to use.
   - title – words to display if the user hovers over the hot zone for a moment.
3. <img> - just like we learned before, except that it includes the "usemap" attribute. Note the pound sign preceding the name.

Graphic coordinates used in computers are similar to Cartesian coordinates. The first number is the horizontal distance measured in pixels from the left side of the image. The second number is the vertical distance downward (opposite from Cartesian) from the top.

You can specify other shapes than "rect". "circle" has three numbers: horizontal and vertical coordinates the center, and radius. "poly" needs a series of horizontal and vertical coordinates for points. The last point should be the same as the first.

Media

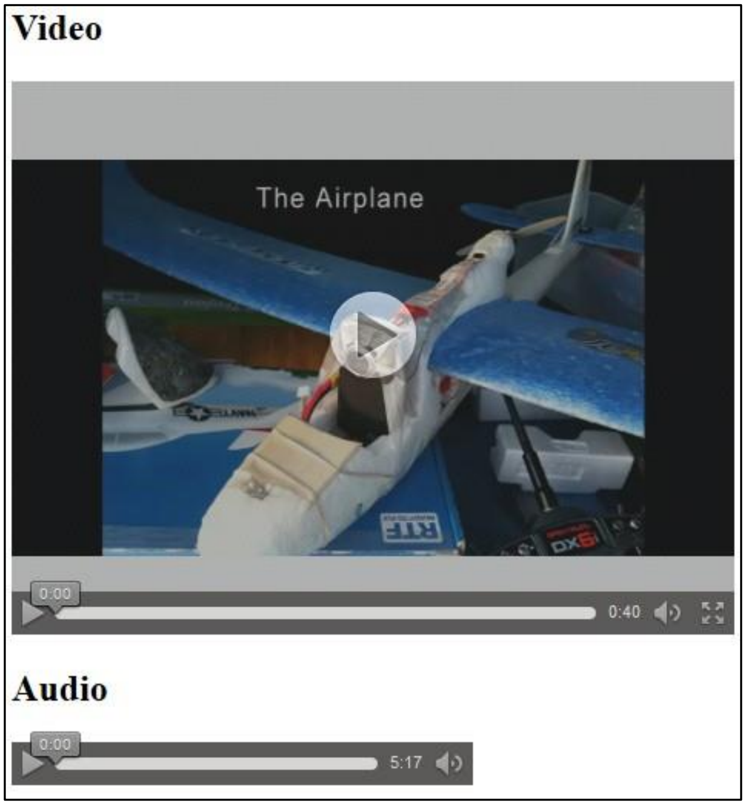
Media generally come under two headings: audio and video. (Every decade or so somebody tries to add smell to electronic media, but for some reason that doesn't play well in the market.) With the arrival of HTML version 5, we at long last have relatively simple and understandable code that can be used for media on the Web.

```
<h2>Video</h2><br />
<video controls="controls"
   width="470" height="360">
   <source src="Landing.mp4" type="video/mp4">
   <source src="Landing.ogv" type="video/ogg">
   <a href="Landing.mp4">Download the video File</a> (.mov)
</video>
<br /><h2>Audio</h2><br />
<audio controls="controls">
    <source src="nb_gg_bremen_mus.mp3"
      type="audio/mpeg">
    <source src="nb_gg_bremen_mus.ogg"
      type="audio/ogg">
    <a href="nb_gg_bremen_mus.mp3">
      Download the Audio File</a> (mp3)
</audio>
```

In the example to the right, we'll use the <video> and <audio> tags for this.

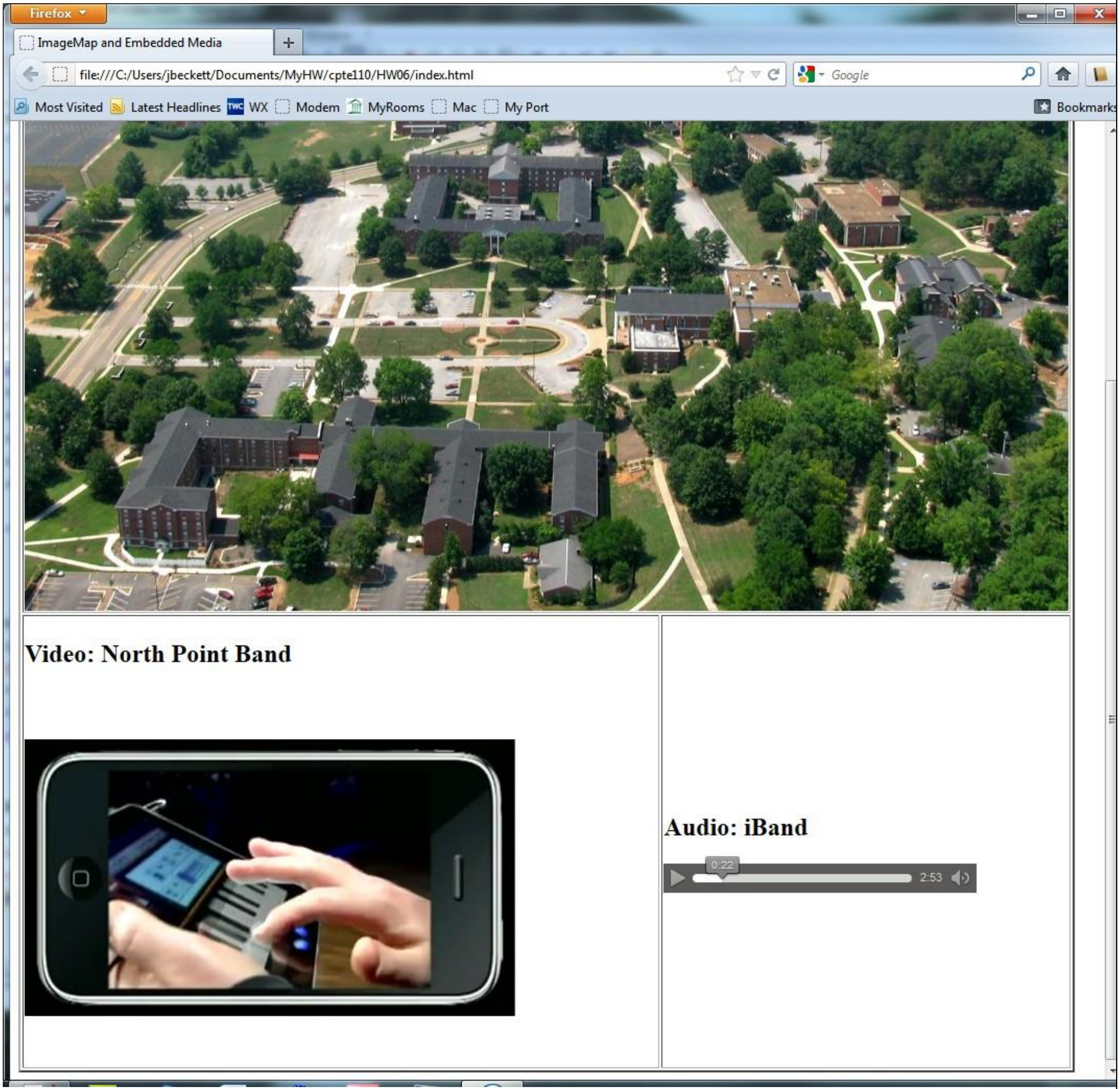
- We specify that we want the user to have controls they can use. For the video we also specify the size of the window.
- We specify two different sources. What I have done is made converted copies as needed to have version of the media file that would play on anything. I use an online converter so I don't have to install any software: http://audio.online-convert.com/convert-to-ogg
- In case no supported file is found for the browser, we have the file itself available for download in an anchor tag.

This code segment is shows up in a browser like this:



HW06: Create a page that:

- Implements an image map with links to Web pages for various buildings on campus
- Demonstrates video and audio using HTML 5.



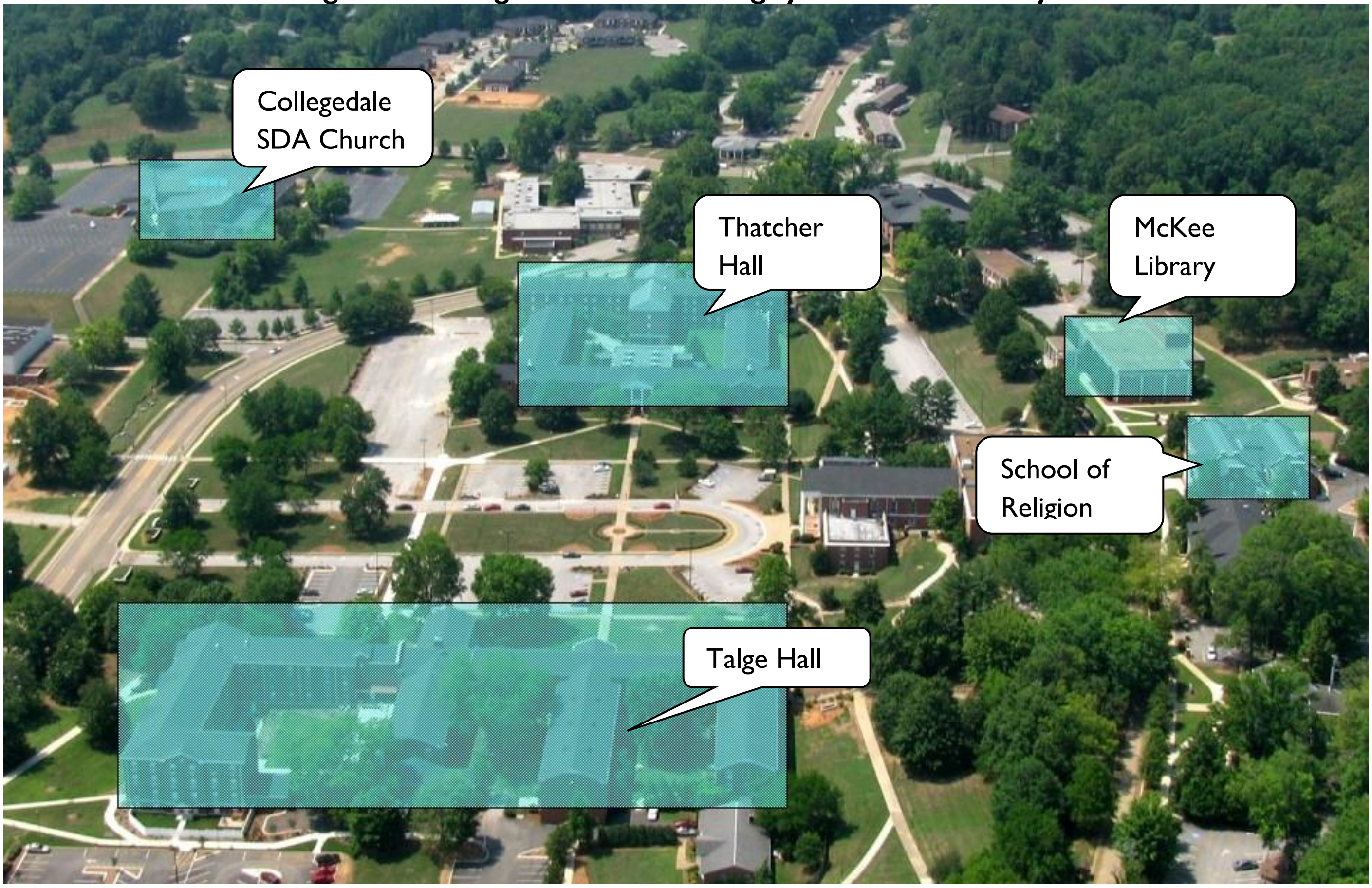
Use a table to structure this page. The campus picture is in a <td> item with colspan="2" and a width of 1000. Hint: Look at http://comprehending.org for a coordinate finder for this assignment and save some time!

For the image map, I strongly suggest that you pull up the source for the example on the support Website and read it carefully. You'll save a lot of time on this assignment! Using the code from this example is OK, but remember: You must adapt it for your own assignment.

Image Map: Use the image file included on the support Web site. Set up hot zones for the following five buildings, linking them to their Web sites. You may use a different aerial photograph and set of sites, but this list will work.

Building	**Web site**
McKee Library	https://www.southern.edu/library
Talge Hall	https://www.southern.edu/talge
Thatcher Hall	https://www.southern.edu/thatcher
School of Religion	https://www.southern.edu/religion
Collegedale SDA Church	http://www.collegedalechurch.com/

Here's a diagram showing where the buildings you need to identify are located.



Lesson 7 – CSS Part 1 – Styling Elements

The next two lessons focus on CSS. Bear in mind that we'll just be scratching the surface. There are huge books on CSS.

Instead of writing our lessons 7 and 8, I'm referring you to some very well-written material that formerly appeared at html.net. We're using their lessons 1-7 for our lesson 7, and their lessons 8-15 for our lesson 8. You'll find links to these on eClass.

However, we have an assignment to do…

HW07: A client whose Website currently has an ugly menu system has seen a demo that uses CSS to do it nicer. In particular, she likes the "dock" look in which whatever item your mouse cursor is over gets bigger because it reminds her of her beloved Macintosh. Your task is to integrate the menu code used in the demo so that the site looks more current.

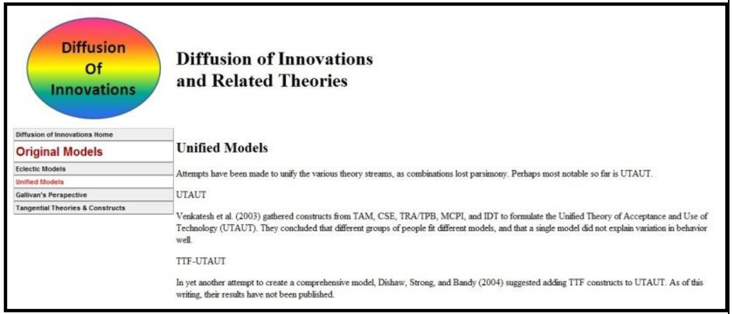
Note: the html.net tutorial #2 describes three ways of linking in CSS code. The demo code you're trying to integrate uses their method #2. You'll be using their method #3.

You'll need two things:

- The former Website
- The demo code

Here's how to do it:

1. Read the code in the demo to see what it does and how to use it.
2. Extract the former Website code into your HW07 folder, and copy the demo code into it as well.
3. Save a copy of the demo code as "menu.css".
4. Remove everything except the styles from menu.css.
5. Work on index.html:
   a. Use "link rel" to reference menu.css.
   b. Change the menu items so that they are in unordered-list (<ul>) format, using the menu CSS.
   c. Add a "thisone" entry to the "home" menu.
   d. Upload the site and verify index.htm to make sure it works properly.
6. One final adjustment to menu.css: Adjust the width to fit better with the logo above.
7. Work through the other documents in the site, making the same changes. Note that "thisone" will be on a different item for each document.
8. Upload again.
9. Go back and check it all out to make sure everything works.



Diffusion of Innovations and Related Theories

Unified Models

Attempts have been made to unify the various theory streams, as combinations lost parsimony. Perhaps most notable so far is UTAUT.

UTAUT

Venkatesh et al. (2003) gathered constructs from TAM, CSE, TRA/TPB, MCPI, and IDT to formulate the Unified Theory of Acceptance and Use of Technology (UTAUT). They concluded that different groups of people fit different models, and that a single model did not explain variation in behavior well.

TTF-UTAUT

In yet another attempt to create a comprehensive model, Dishaw, Strong, and Bandy (2004) suggested adding TTF constructs to UTAUT. As of this writing, their results have not been published.

The example shown is the "Unified Models" page. Note that the menu button for this page is colored. The cursor was hovering over "Original Models" at the time this screen-shot was made, so that selection was magnified.

If Sominex isn't working to get you sleepy when you need it to, the whole story is at http://computing.southern.edu/jbeckett/dissertation/.

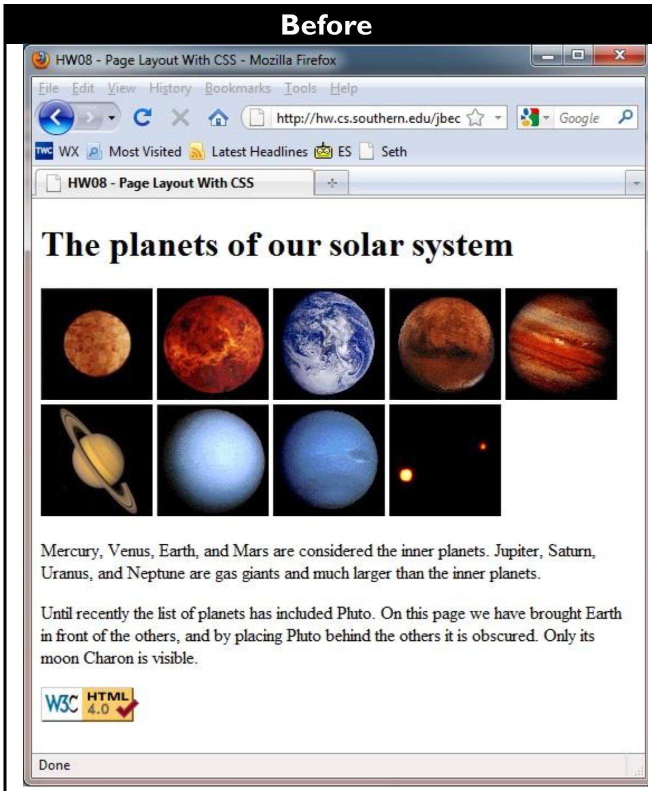
Lesson 8 – CSS Part 2 – Layout With Style

Instead of writing our lessons 7 and 8, I'm referring you to some very well-written material at http://www.html.net/tutorials/css/. We will use their lessons 8-15.

However, we have an assignment to do…

HW08: CSS Layout
Obtain a set of images and index.html from the support site. Use the HTML provided on the class Web site – without any changes. Place the images in an "images" folder. Your task is to create an external style sheet (hw08.css) that will create the final effect desired. Here's how the page will look before and after CSS is applied:

Before


HW08 - Page Layout With CSS - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://hw.cs.southern.edu/jbec
Google
WX Most Visited Latest Headlines ES Seth
HW08 - Page Layout With CSS
Done

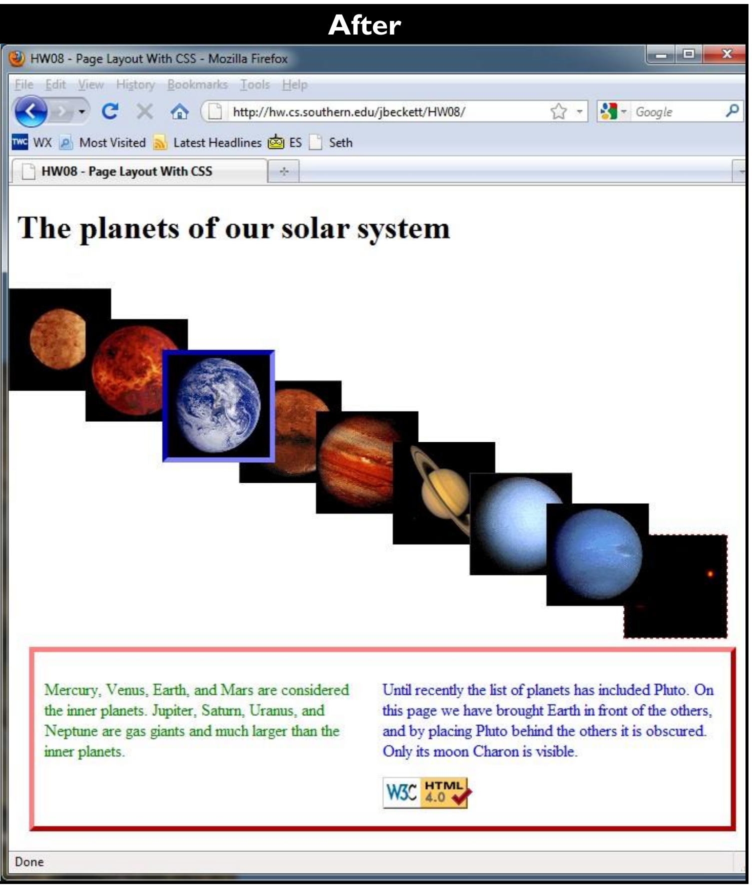
The planets of our solar system

Mercury, Venus, Earth, and Mars are considered the inner planets. Jupiter, Saturn, Uranus, and Neptune are gas giants and much larger than the inner planets.

Until recently the list of planets has included Pluto. On this page we have brought Earth in front of the others, and by placing Pluto behind the others it is obscured. Only its moon Charon is visible.

W3C HTML 4.0

After


HW08 - Page Layout With CSS - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://hw.cs.southern.edu/jbeckett/HW08/
Google
WX Most Visited Latest Headlines ES Seth
HW08 - Page Layout With CSS
Done

The planets of our solar system

Mercury, Venus, Earth, and Mars are considered the inner planets. Jupiter, Saturn, Uranus, and Neptune are gas giants and much larger than the inner planets.

Until recently the list of planets has included Pluto. On this page we have brought Earth in front of the others, and by placing Pluto behind the others it is obscured. Only its moon Charon is visible.

W3C HTML 4.0

Style rules to create:

Rule name	Purpose
#planet1	Locate at the absolute position 0 pixels from left, 100 pixels down from the top.
#planet2	Locate at the absolute position 75 pixels from left, 130 pixels down from the top.
#planet3	Locate at the absolute position 150 pixels from left, 160 pixels down from the top.
And so on through #planet9 – increasing pixels from the left by 75 each time and from the top by 30 each time.	
Note: Earth (planet 3) and Pluto (planet 9) must have Z-indexes that place them properly (Earth on top, Pluto	

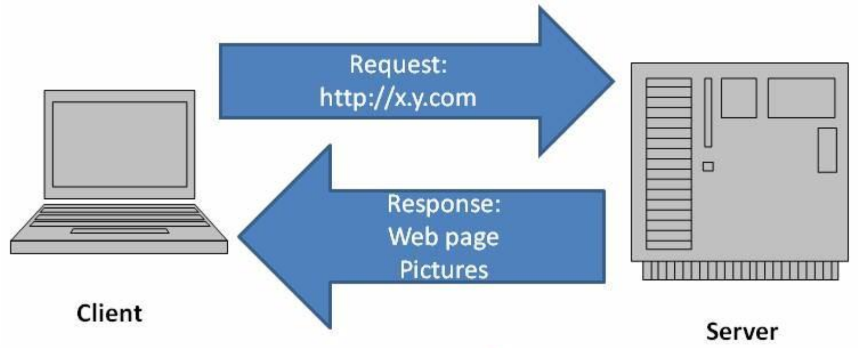
behind). They also need borders. Earth gets a thick blue border of the “inset” style, and Pluto gets a thin red border in “dashed” style.	
.endofpage	This will be a box 660 pixels wide positioned at an absolute location 20 pixels from the left and 450 pixels down from the top. It will have a height of 150 pixels. Pad it 10 pixels all around. Give it a red border that is thick, in “outset” style.
#column1	Use the left 45% of the page and set the text color to green.
#column2	Use the next 50% of the page and set the text color to blue.

Lesson 9 – Forms

So far in this class, we've focused on presenting information to the Website visitor. Everything has gone from our hard drive to the visitor's screen. But that is not the Internet as we know it: a place where people interact and where business is done. The key is information flow from the Website visitor into our Web application. And the way it's done is with Web forms.

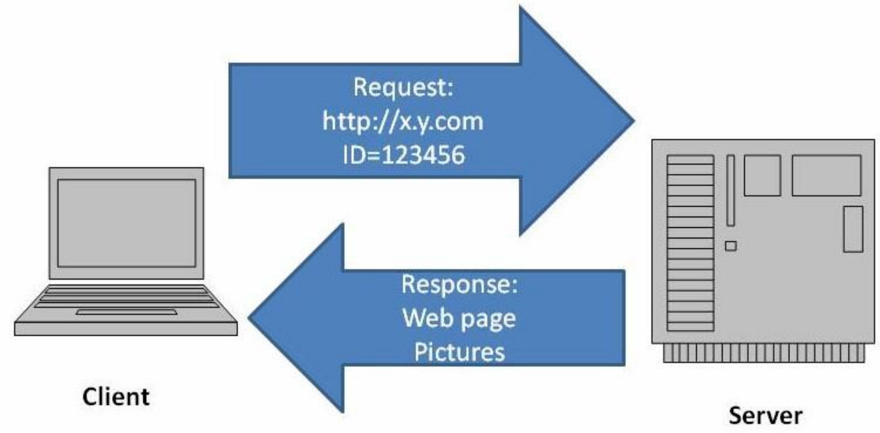
⚠ If you send information to a Website, anything you send might be intercepted (possibly by someone who does not like you or wants to steal from you) unless you encrypt the information. The easiest way to make sure you are using encryption is to look at the url bar on your browser for "`https://`" instead of "`http://`". Bear in mind that this does not guarantee your message is going where you thought it should. For instance, you might have been emailed a url that looked a lot like you bank's address but was a hacker site in Eastern Europe.

To understand how this works, review the basic architecture of the Web:



A visitor to your site sends a request for a page, either by typing in its url or by clicking on a link. In response, the server sends what was requested. We call this a "static" Web page.

When a form is used, something new happens: Information is sent *as a part of the request.* The HTML you have learned so far never "sees" this information, but with special code called "server-side" we can actually use the information. This class does not include server-side programming, so I'll provide you with a server-side page that simply displays the information. It's on the support Website, and it is named "process.php". I've added a ".txt" extension on the Website for technical reasons. What you need to do it upload it to your site on the server, renaming it "process.php" so it will work properly.



The new architecture is called "**dynamic**" – meaning that the content that is sent changes depending on the information sent to it.

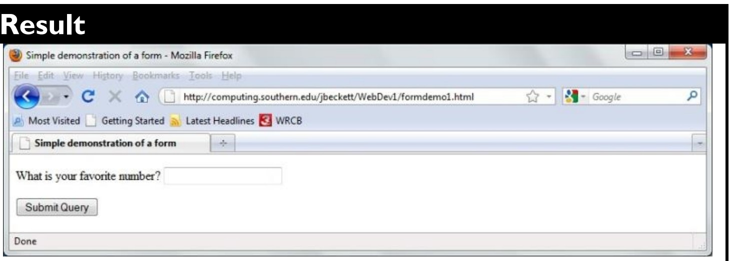
A very simple dynamic Web page

Our first page with a form will simply ask someone's favorite number. I've highlighted the part of interest to us here, but I'm including the entire page.

Code

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=utf-8" />
<title>Simple demonstration of a form</title>
</head><body>
<form action="process.php">
<p>What is your favorite number?
  <input type="text" name="favnum" /></p>
<input type="submit" />
</form>
</body>
```

Result



Simple demonstration of a form - Mozilla Firefox

http://computing.southern.edu/jbeckett/WebDev1/formdemo1.html

What is your favorite number?

File Edit View History Bookmarks Tools Help
Google
Most Visited
Getting Started
Latest Headlines
WRCB
Simple demonstration of a form
Submit Query
Done

The <form> tag defines the area used for input from the Website visitor. Interesting parts of this little demo:

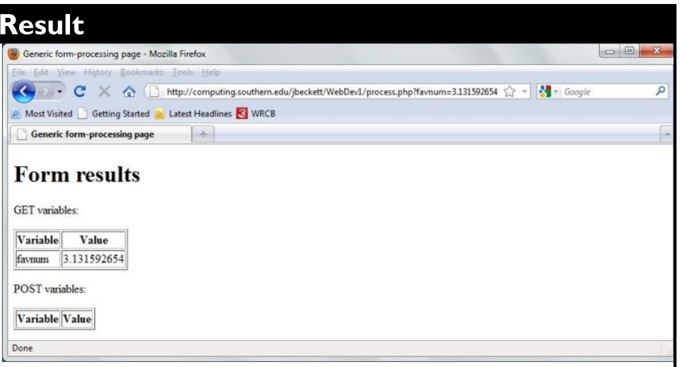
- The action attribute of the <form> tag indicates what will happen when the visitor clicks the Submit button. It is the name of a file. In effect, the submit button acts like a link to the page identified by the action attribute.
- The first <input> tag is type "text", meaning that some form of keyboard input is expected here.
- The second <input> tag is type "submit", meaning this is a button used to submit the form to the server.

When the submit button is clicked, the browser sends a new URL to the server. It includes the data on the URL line, after a question mark.

Code

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Generic form-processing page</title>
<meta http-equiv="Content-Type"
content="text/html; charset=iso-8859-1">
</head>
<body>
<h1>Form results</h1>
<p>GET variables:</p>
<table
border="1"><tr><th>Variable</th><th>Value</th></t
r>
<?php
foreach($_GET as $p => $q) {
print("<tr><td>" . $p . "</td><td>" . $q .
"</td></tr>");
}?></table><p>POST variables:</p><table
border="1"><tr><th>Variable</th><th>Value</th></t
r>
<?php
foreach($_POST as $p => $q) {
print("<tr><td>" . $p . "</td><td>" . $q .
"</td></tr>");
}?></table></body></html>
```

Result



Generic form-processing page - Mozilla Firefox

http://computing.southern.edu/jbeckett/WebDev1/process.php?favnum=3.131592654

Form results

GET variables:

Variable	Value
favnum	3.131592654

POST variables:

Variable	Value

File Edit View History Bookmarks Tools Help
Google
Most Visited
Getting Started
Latest Headlines
WRCB
Generic form-processing page
Done

We'll use the same process.php code for examples below but won't show it, because it is unchanged.

I'm including the code on the left so that you'll know it there. But relax: we aren't learning any server-side code in this class. Suffice it to say that process.php simply shows whatever data was sent to it, in this case on the URL line.

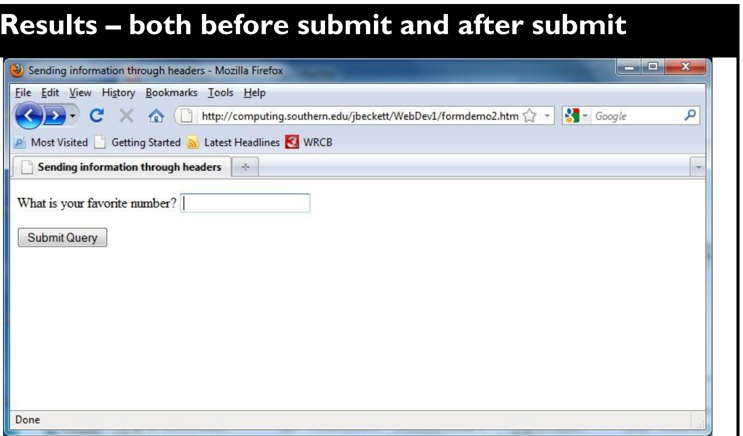
Sending information in the header instead of the URL line (GET versus POST)

In that last example we specified `method="GET"`, which uses the URL line to send the information. Sometimes that isn't what we want to do. Possibly we don't want the user to be able to bookmark that URL line and re-invoke the same function (because the information has changed). Or we don't want the user to even see the information being sent. We can do that by changing to `method="POST"`. Like this:

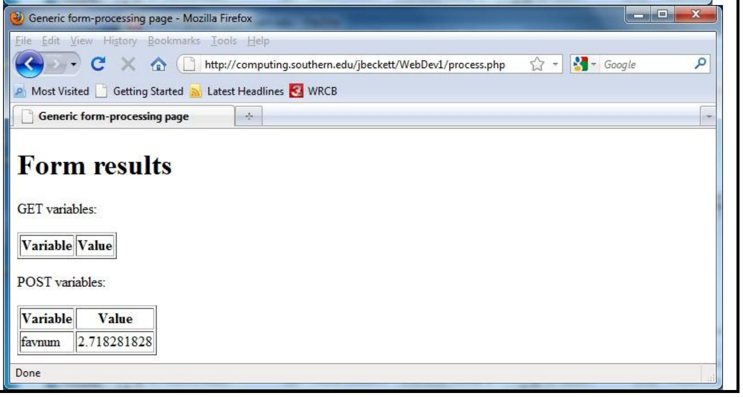
HTML Code
(server-side program is the same as above)

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=utf-8" />
<title>Sending information through
headers</title>
</head><body>
<form action="process.php" method="post">
<p>What is your favorite number?
  <input type="text" name="favnum" /></p>
<input type="submit" />
</form>
</body></html>
```

Results – both before submit and after submit


Sending information through headers - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://computing.southern.edu/jbeckett/WebDev1/formdemo2.htm
Google
Most Visited Getting Started Latest Headlines WRCB
Sending information through headers
Submit Query
Done

What is your favorite number?


Generic form-processing page - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://computing.southern.edu/jbeckett/WebDev1/process.php
Google
Most Visited Getting Started Latest Headlines WRCB
Generic form-processing page
Done

Form results

GET variables:

Variable	Value

POST variables:

Variable	Value
favnum	2.718281828

Referring to those diagrams at the beginning of this lesson, the information is being sent as part of the header for the request. It does not show up on the URL line but the processing program gets it and may use it.

Why would one use GET or POST?

- GET is handy for search arguments. If you search Google and find an interesting result, you can copy the URL line in all its ugliness and send a copy to someone else. Then they can click on it and get the same result.

- POST is handy for hiding information from the user. If they enter a password, you don't want to show it on the URL line. Bear in mind this gives only weak security since the password is still being sent over the Internet. It would be possible, for instance, to write a browser add-in that gives a pop-up showing all the POST data.

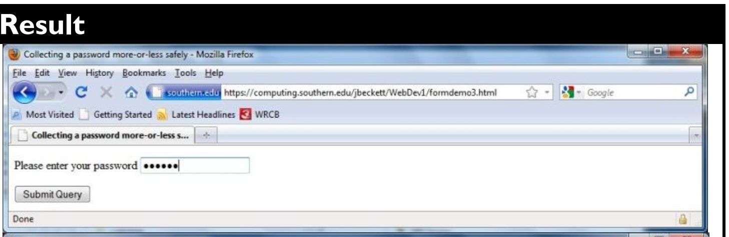
Passwords

So, does this make our form data secure? Unfortunately not If we're talking on a wireless link our information can be captured by a hacker standing by listening in. So we should really use https// instead of http:// if we want to keep the information private. Here's a page that collects a password, and we access it using https://

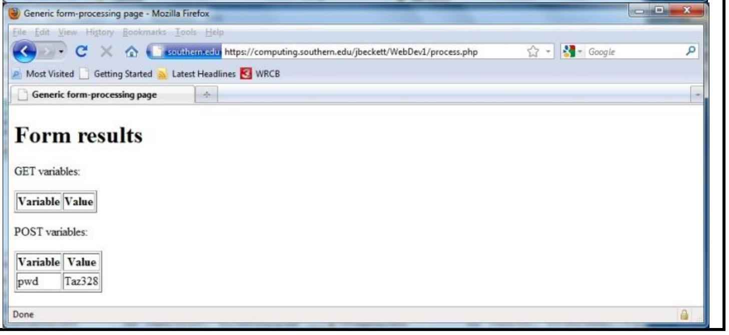
Code

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=utf-8" />
<title>Collecting a password more-or-less
safely</title>
</head><body>
<form action="process.php" method="post">
<p>Please enter your password
  <input type="password" name="pwd" /></p>
<input type="submit" />
</form>
</body></html>
```

Result


Collecting a password more-or-less safely - Mozilla Firefox
File Edit View History Bookmarks Tools Help
southern.edu https://computing.southern.edu/jbeckett/WebDev1/formdemo3.html
Google
Most Visited Getting Started Latest Headlines WRCB
Collecting a password more-or-less s...
Submit Query
Done

Please enter your password ••••••


Generic form-processing page - Mozilla Firefox
File Edit View History Bookmarks Tools Help
southern.edu https://computing.southern.edu/jbeckett/WebDev1/process.php
Google
Most Visited Getting Started Latest Headlines WRCB
Generic form-processing page
Done

Form results

GET variables:

Variable	Value

POST variables:

Variable	Value
pwd	Taz328

The password is not echoed, but the browser shows bullet characters so the visitor can determine how many characters they have typed. So in this case we've done two things to limit our exposure to bad people:

1. Used the type="password" attribute to prevent the password from being seen while it was typed.
2. Used https:// instead of http:// to encrypt the connection, in case somebody is snooping.

Note that each of these two methods protects your site against a specific group of attackers. The first protects against shoulder surfers, and the second protects against hackers you don't see. Use **both** methods.

A common error in the assignment for this chapter is to use a `<form> tag` for each of the items in your form. For example:

Wrong

```
<form action="process.php"
  method="POST">
<input type="text" name="name" />
</form>
<form action="process.php"
  method="POST">
<input type="submit" />
</form>
```

Right

```
<form action="process.php"
  method="POST">
<input type="text" name="name" />
</form>
<input type="submit" />
</form>
```

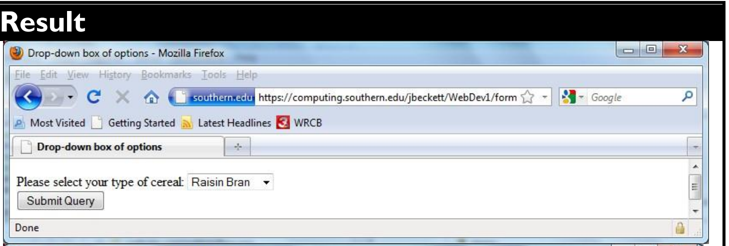
Drop-down boxes

We could collect all sorts of data using <input> - but that allows our site visitors to type whatever they wish. A common way to guide them is using a drop-down box. Like this:

Code

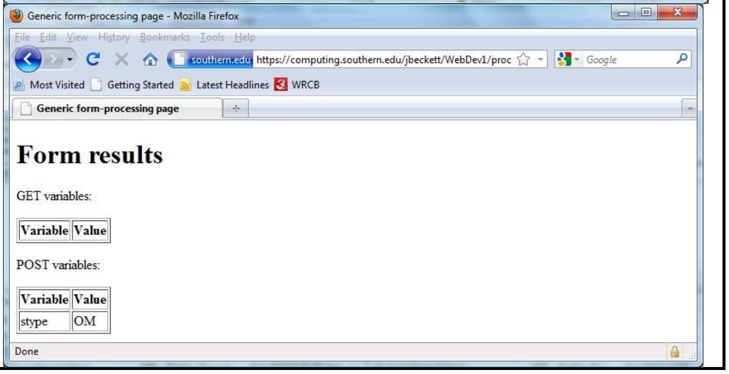
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=utf-8" />
<title>Drop-down box of options</title>
</head><body>
<form action="process.php" method="post">
<p>Please select your type of cereal:
  <select name="stype">
    <option value="CF">Corn Flakes</option>
       <option value="RB"
selected="selected">Raisin Bran</option>
       <option value="OM">Hot Oatmeal</option>
  </select>
  <br />
  <input type="submit" />
</form>
</body></html>
```

Result


Drop-down box of options - Mozilla Firefox
File Edit View History Bookmarks Tools Help
southern.edu https://computing.southern.edu/jbeckett/WebDev1/form
Google
Most Visited Getting Started Latest Headlines WRCB
Drop-down box of options
Done

Please select your type of cereal: Raisin Bran

Submit Query


Generic form-processing page - Mozilla Firefox
File Edit View History Bookmarks Tools Help
southern.edu https://computing.southern.edu/jbeckett/WebDev1/proc
Google
Most Visited Getting Started Latest Headlines WRCB
Generic form-processing page
Done

Form results

GET variables:

Variable	Value

POST variables:

Variable	Value
stype	OM

In this case I (the site visitor) clicked the arrow to the right of the box, selected "Hot Oatmeal", and clicked the submit button.

The <select> and <option> tags work together. <select> indicates that this is a group of options that will appear in a drop-down box There are three attributes used in this to control our data:

- We define the name that will be used to pass the information in the <select> tag. Each of possible selections is indicated with an <option> tag.
- We use the "value" attribute in the <option> tag to indicate the code that we'll send to the processing page for this particular selection.
- We may optionally use the "selected" attribute in one of the <option> tags to indicate the default selection. If we don't do this, the default will be the first option.

⚠ Note that in a <select> pull-down, the name of the item to be used by the processing program is in the <select>, not the <option>

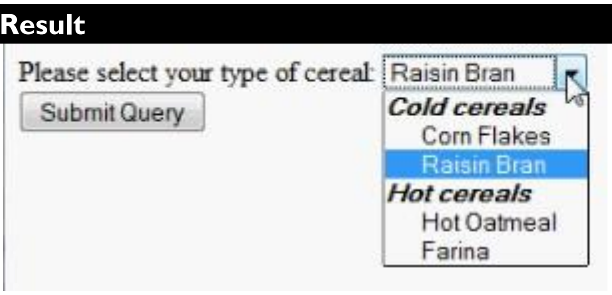
Grouping options in a drop-down box

If your drop-down box is too large, people might find it difficult to find what they want. For the next few examples we'll only include the specific code to save paper:

Code

```
<p>Please select your type of cereal:
  <select name="stype">
    <optgroup label="Cold cereals">
      <option value="CF">Corn Flakes</option>
         <option value="RB"
selected="selected">Raisin Bran</option>
        </optgroup>
        <optgroup label="Hot cereals">
          <option value="OM">Oatmeal</option>
          <option value="FA">Farina</option>
        </optgroup>
  </select>
```

Result



Please select your type of cereal: Raisin Bran

The various sections of the list are identified with boldfaced/italicized group names (which themselves cannot be selected.)

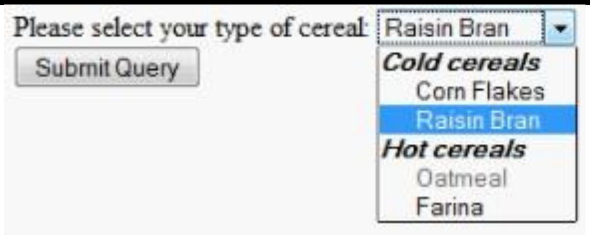
Disabling options

Speaking of not being selectable: You can also set up options so they are "grayed-out" – a way of letting your site visitor know that this option is available sometimes, but not now. If we're out of oatmeal, we could do this:

Code

```
  <select name="stype">
    <optgroup label="Cold cereals">
      <option value="CF">Corn Flakes</option>
         <option value="RB"
selected="selected">Raisin Bran</option>
        </optgroup>
        <optgroup label="Hot cereals">
          <option disabled="disabled"
value="OM">Oatmeal</option>
          <option value="FA">Farina</option>
        </optgroup>
  </select>
```

Result



Please select your type of cereal: Raisin Bran

Note that the result is identical to the previous example, except that "Oatmeal" is grayed-out and thus cannot be selected.

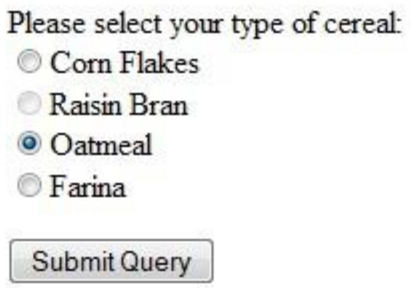
Radio buttons

As an alternative to a drop-down box, you could show the choices in a group of radio buttons. Let's re-do our cereal example using them:

Code

```
<p>Please select your type of cereal:<br />
  <input type="radio" name="stype" value="CF" />Corn Flakes<br
/>
  <input type="radio" name="stype" value="RB"
disabled="disabled" />Raisin Bran<br />
  <input type="radio" name="stype" value="OM" checked="checked"
/>Oatmeal<br />
  <input type="radio" name="stype" value="FA" />Farina<br />
</p>
```

Result



Please select your type of cereal:

○ Corn Flakes
○ Raisin Bran
● Oatmeal
○ Farina

Since all the buttons have the same name attribute "stype", they will be considered a group and the site visitor can only check one. We've grayed-out Raisin Bran this time for illustration, and used the "checked" attribute to pre-select the healthiest alternative of this group.

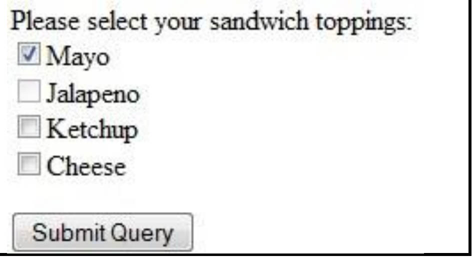
Check boxes

Sometimes choices are not mutually exclusive. How about a sandwich shop's toppings:

Code

```
<p>Please select your sandwich toppings:<br />
  <input type="checkbox" name="MA" checked="checked" value="MA"
/>Mayo<br />
  <input type="checkbox" name="JP" value="JP"
disabled="disabled" />Jalapeno<br />
  <input type="checkbox" name="KE" value="KE" />Ketchup<br />
  <input type="checkbox" name="CH" value="CH" />Cheese<br />
</p>
```

Result



Please select your sandwich toppings:

☑ Mayo
□ Jalapeno
□ Ketchup
□ Cheese

The checkboxes are now square instead of round, to give our visitor a hint that the selections are not mutually exclusive.

In this case we pre-select "Mayo" using the "checked" attribute, since just about everybody likes it. And alas, we're out of peppers. We can also provide ketchup and cheese. (Yes, I know the shop will never make it with so few options but we're trying to save complicated code on this page. I've noticed over the years that real-life projects are generally much more complicated than you thought at first, even if they seemed pretty simple.)

Another critical difference here: Since we need separate information out of each of those checkboxes, they are named differently.

Text Boxes

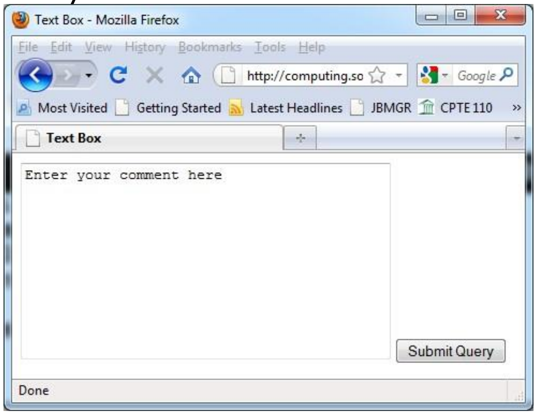
Sometimes it takes more than one line. A text box is used for this:

Code

```
<form action="process.php" method="get">
  <textarea name="comment"
    cols="40" rows="10" />
    Enter your comment here
  </textarea>
    <input type="submit" />
</form>
```

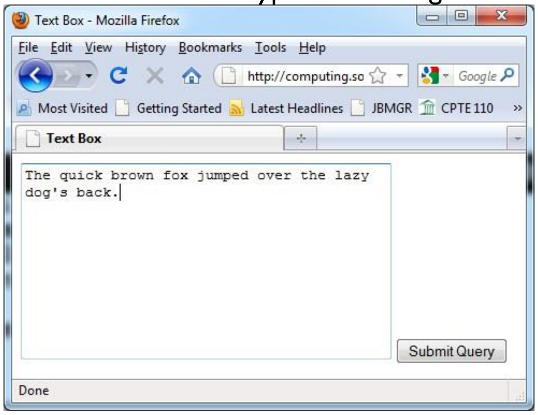
Results

Initially it looks like this:



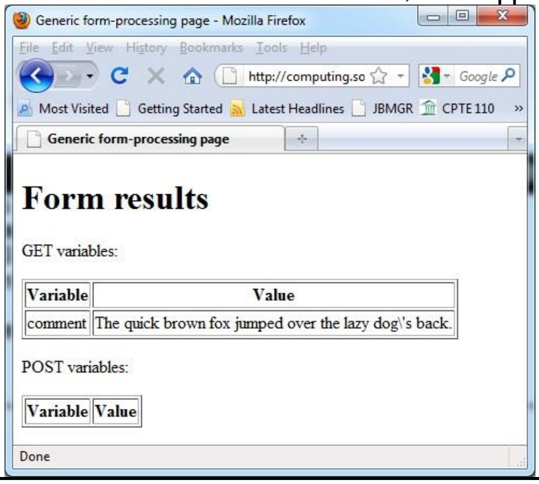
```
Enter your comment here
```

Then the site visitor types something like this:



```
The quick brown fox jumped over the lazy
dog's back.
```

When the click the submit button, this happens:


Generic form-processing page - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://computing.so
Google
Most Visited
Getting Started
Latest Headlines
JBMGR
CPTE 110
Generic form-processing page
Done

Form results

GET variables:

Variable	Value
comment	The quick brown fox jumped over the lazy dog\'s back.

POST variables:

Variable	Value

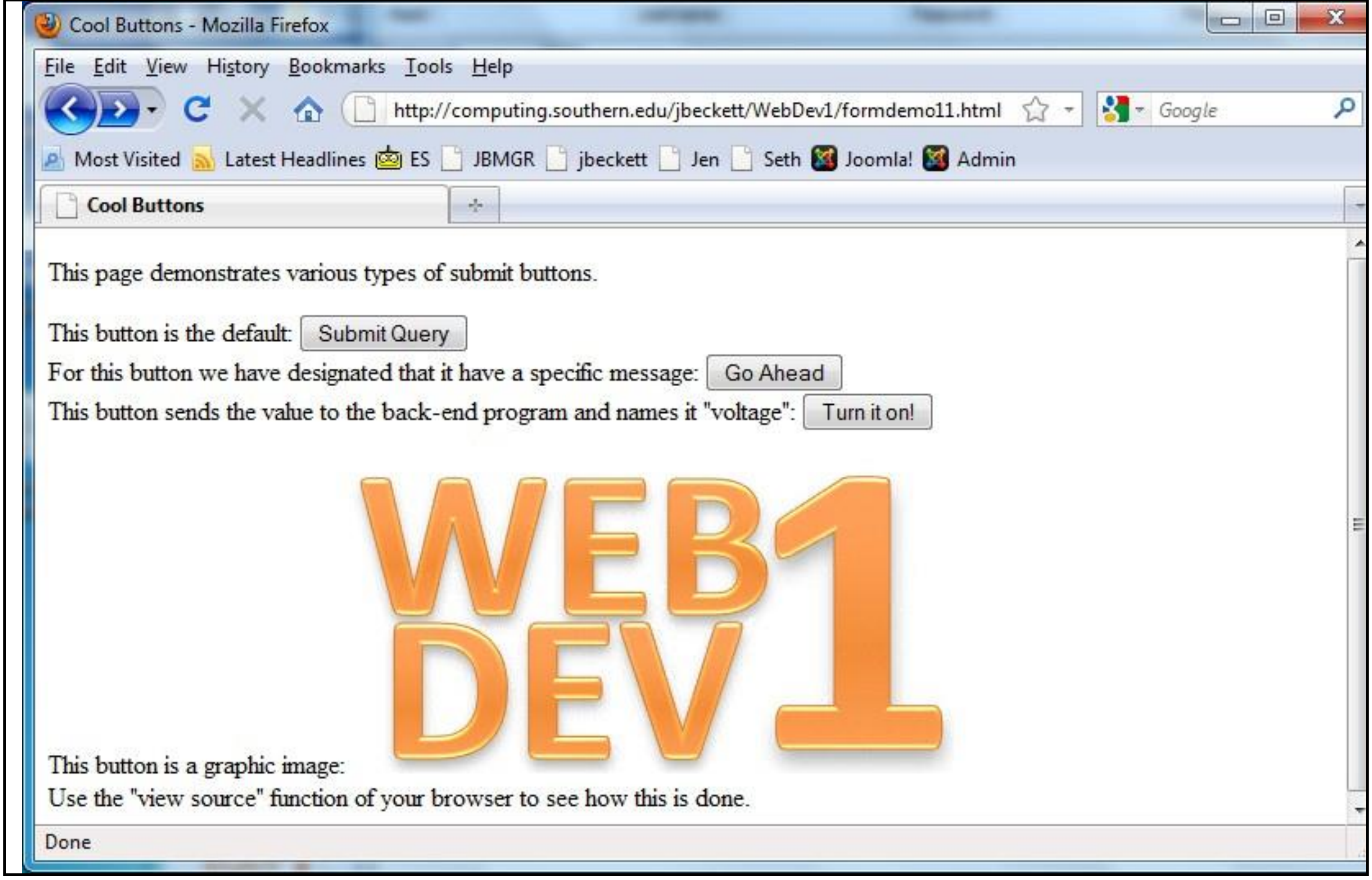
Submit Buttons

The submit button can have words you specify, or be a graphic. And you may have multiple submit buttons - each of which passes a specific message along to the receiving program. Here's how:

Code

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Cool Buttons</title>
</head><body>
<p>This page demonstrates various types of submit buttons.
<form action="process.php" method="get">
This button is the default:
  <input type="submit" /><br />
For this button we have designated that it have a specific message:
  <input type="submit" value="Go Ahead" /><br />
This button sends the value to the back-end program and names it "voltage":
  <input type="submit"
     name="voltage" value="Turn it on!" />
<br /><br />
This button is a graphic image:
  <input type="image" src="WebDev1Logo.gif"
     name="ClickedLogo" value="Yess" /><br />
Use the "view source" function of your browser to see how this is done.
</form>
</body>
```

Results


Cool Buttons - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://computing.southern.edu/jbeckett/WebDev1/formdemo11.html
Google
Most Visited
Latest Headlines
ES
JBMGR
jbeckett
Jen
Seth
Joomla!
Admin
Cool Buttons
WEB DEV 1
Done

This page demonstrates various types of submit buttons.

This button is the default: Submit Query

For this button we have designated that it have a specific message: Go Ahead

This button sends the value to the back-end program and names it "voltage": Turn it on!

This button is a graphic image:

Use the "view source" function of your browser to see how this is done.

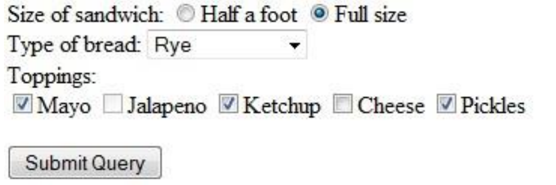
Pulling it together

All this talk about food has me hungry and it's suppertime as I write this. So let's put together an order page for a sandwich shop. Since I'm a vegetarian, we won't bother with meat selection.

Code

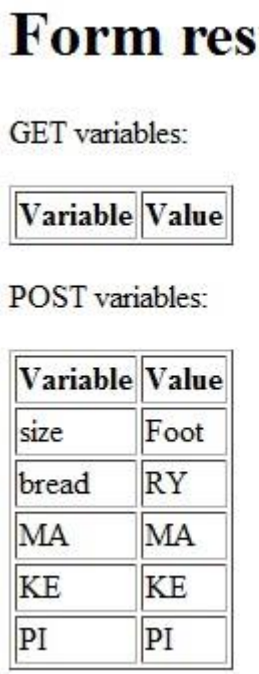
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Ordering a sandwich</title>
</head><body>
<form action="process.php" method="post">
<p>Size of sandwich:
    <input type="radio" name="size" value="Six" />Half a
foot
       <input type="radio" name="size" value="Foot"
/>Full size<br />
   Type of bread:
    <select name="bread">
         <option value="WH">Wimpy-White</option>
         <option selected="selected" value="WW">Whole-
wheat</option>
         <option value="RY">Rye</option>
       </select><br />
Toppings:<br />
  <input type="checkbox" name="MA" checked="checked"
value="MA" />Mayo
  <input type="checkbox" name="JP" value="JP"
disabled="disabled" />Jalapeno
  <input type="checkbox" name="KE" value="KE" />Ketchup
  <input type="checkbox" name="CH" value="CH" />Cheese
  <input type="checkbox" name="PI" value="PI"
/>Pickles<br />
</p>
  <input type="submit" />
</form>
</body></html>
```

Results (two screens shown)



Size of sandwich: ○ Half a foot ● Full size
Type of bread: Rye ▾
Toppings:
☑ Mayo □ Jalapeno ☑ Ketchup □ Cheese ☑ Pickles

Submit Query



Form results

GET variables:

Variable	Value

POST variables:

Variable	Value
size	Foot
bread	RY
MA	MA
KE	KE
PI	PI

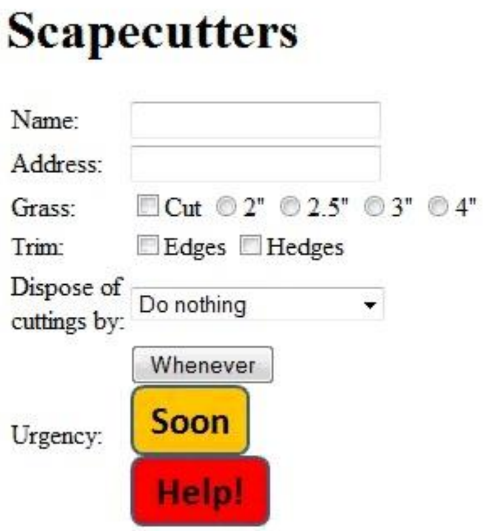
HW09: Create a page for use in ordering landscaping service. All the good names were already taken, so they call themselves “Scapecutters.”

Use the POST method in your <form> tag.

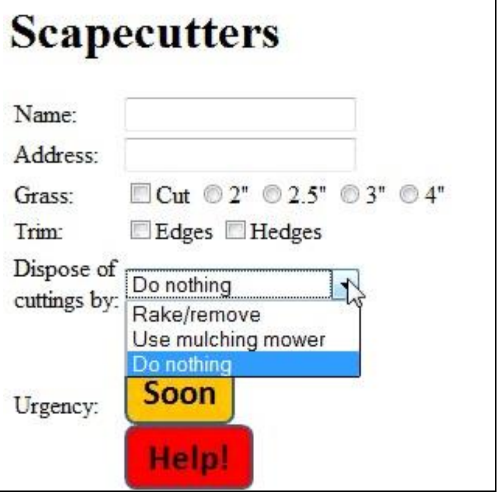
You’ll be collecting the following information:

- Name and address, both text fields.
- A checkbox for whether or not to cut the grass, and a set of radio buttons for the height as shown. (If this were a class that included serious JavaScript we’d also have you darken the height selectors if they don’t want the grass cut.)
- Check boxes for trimming edges of the lawn, and the hedges.
- A drop-down box to select how the cuttings should be handled.
- Three buttons, each of which passes information about the urgency of the request under the name “when”. The graphic buttons are available on the support Website.

The form should send data to process.php. This is a back-end program that’s available on the support Website. It downloads under the name “scapeproc.php.txt” – which you need to rename “process.php”.



The examples shown here illustrate the action of the drop-down list. Note that it has “Rake/remove” as the first option, then mulching, then nothing – but the default should be nothing.



HW09 continued:

For the homework assignment, we'll use an expansion of process.php called scapeproc.php. This name will be used in the "action=" attribute of your <form> tag.

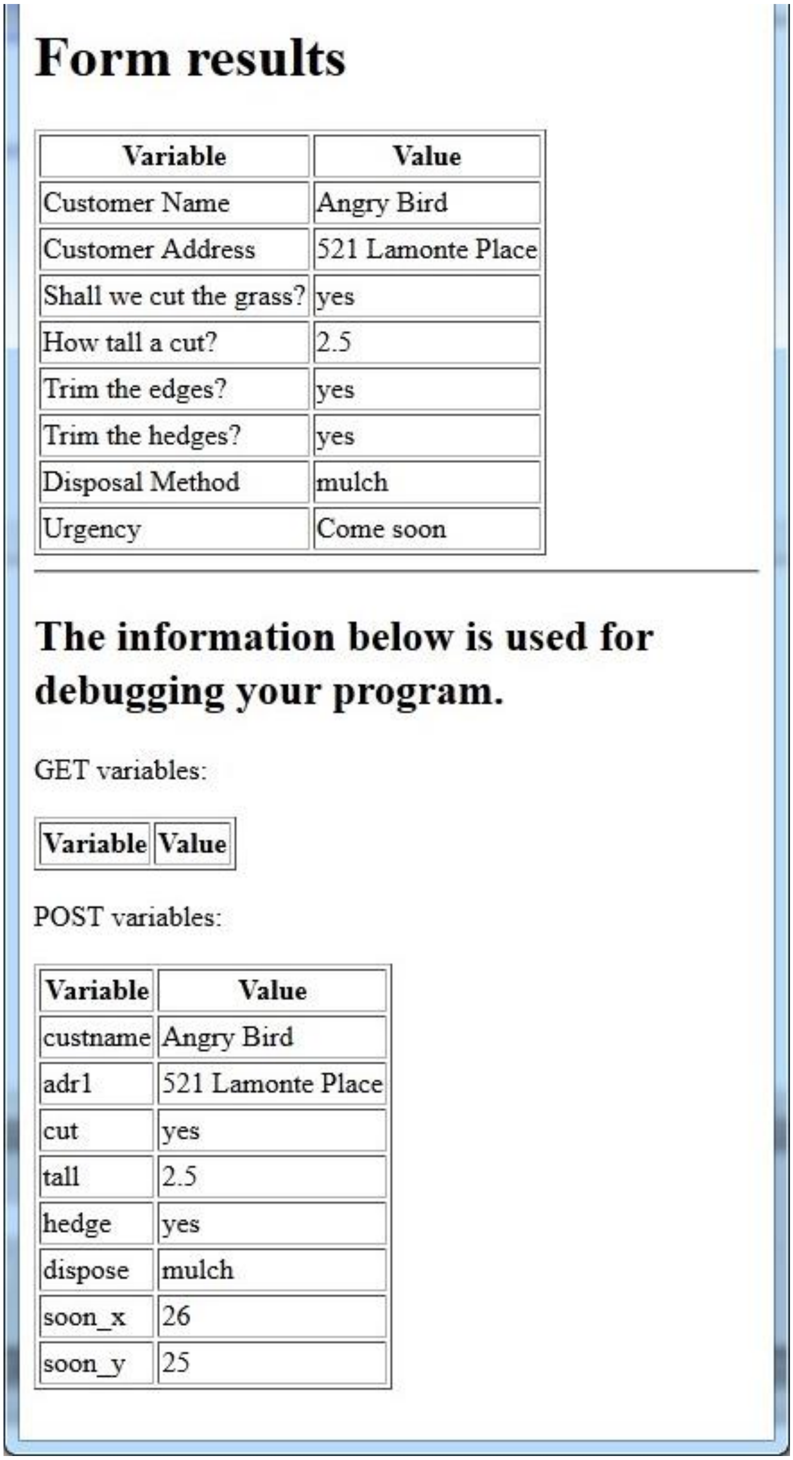
Here is the output of scapeproc.php if the "soon" button is clicked.

Make sure you name your form variables correctly so that the result will work properly. Here are the names you should use in your index.html file (all lower-case letters!)

- custname
- adr1
- cut
- tall
- edge
- hedge
- dispose
- when
- soon
- help

scapeproc.php is written for you – downloadable from the support site as scapeproc.php.txt. You'll have to rename it and upload it. This program has two sections:

- Information that will be shown to Scapecutter's office from the form.
- A section that shows the information coming from your form. If something isn't showing up in the Form Results section, you probably haven't named it correctly – look at the list above.



Form results

Variable	Value
Customer Name	Angry Bird
Customer Address	521 Lamonte Place
Shall we cut the grass?	yes
How tall a cut?	2.5
Trim the edges?	yes
Trim the hedges?	yes
Disposal Method	mulch
Urgency	Come soon

The information below is used for debugging your program.

GET variables:

Variable	Value

POST variables:

Variable	Value
custname	Angry Bird
adr1	521 Lamonte Place
cut	yes
tall	2.5
hedge	yes
dispose	mulch
soon_x	26
soon_y	25

WARNING: By now you should have learned that failure to test your projects by using HTTP:// access in a browser is a very bad idea. But if you've been very careful, you might have gotten away with "testing" projects only on your personal computer. For this project, you absolutely must use HTTP:// in a browser or you are guaranteed to fail. If you don't understand the difference, now is the time to ask your instructor!

Lesson 10 – Content Management Systems - 1

Very little of the content of today's Web is typed by those who write it in HTML. Make no mistake, it's important to know HTML for those times it's needed – and you definitely have more control over your Website if you can use HTML at the right times. But these days most of the content of Websites is kept in databases and generated on-the-fly by systems of programs called Content Management Systems (CMS). Why use a CMS?

1. The structure, visual design, and content are each contributed by different people. Designers aren't accidentally changing the words, people who are in charge of the structure of a Website don't change its look when they change the function, and those who put in the words are carefully controlled (not a bad idea for social networking sites!)
2. With structure, visual design and content in different places, it's relatively easy to change one without re-doing another.
3. The three groups of contributors don't have to know how the other part is done.

⚠ A frequent error on quizzes in this class: not noticing that a CMS stores content data in a database instead of HTML. That's the whole purpose of a CMS: to take advantage of the management capabilities of a database for information that is likely to change.

There are many CMS programs in use. You've probably used Facebook or MySpace. These are CMS programs aimed at social networking. In the corporate world one of the most common is SharePoint from Microsoft. SharePoint is tightly tied into the Microsoft networking scheme, and fits naturally for organizations that build their IT around Microsoft Exchange. And yes, it costs money.

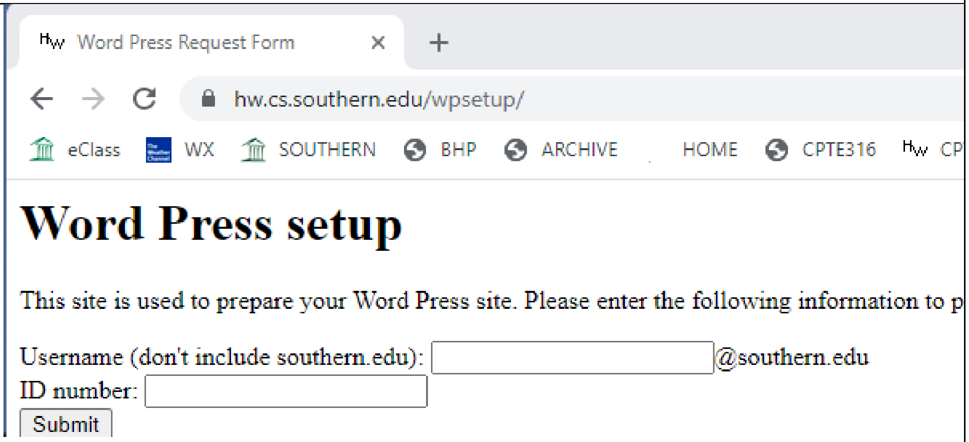
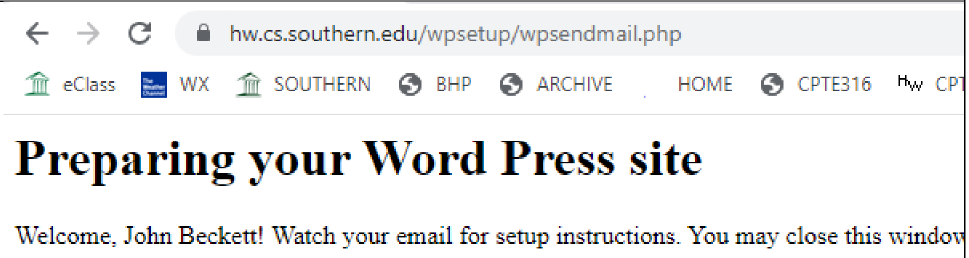
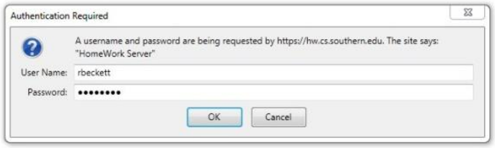
For communities of practice (groups of people united around a common purpose) that are not part of a Microsoft Exchange group, another popular CMS is WordPress, a freely-available open-source package. WordPress runs on Windows, Linux and Mac OS since it is written in PHP – a language that runs on almost any computer. Thousands of Web sites use WordPress to give them the CMS advantage.

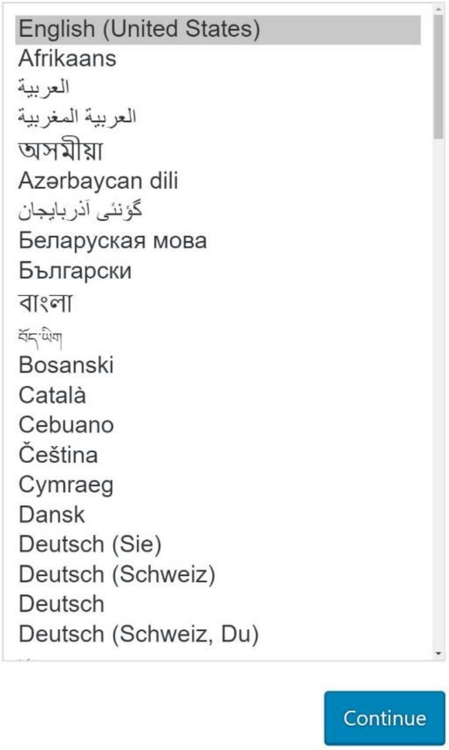
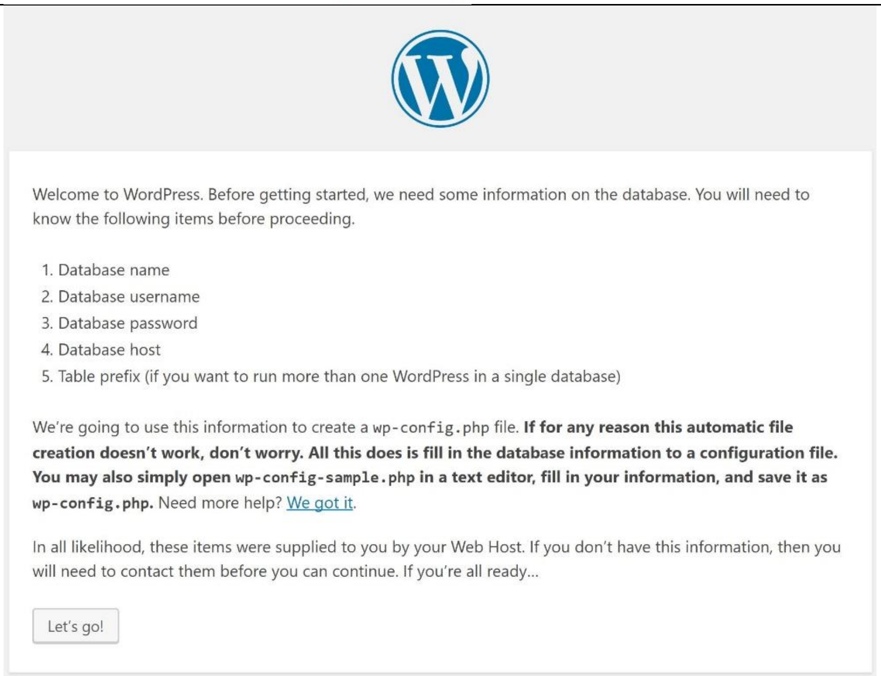
WordPress has three parts:

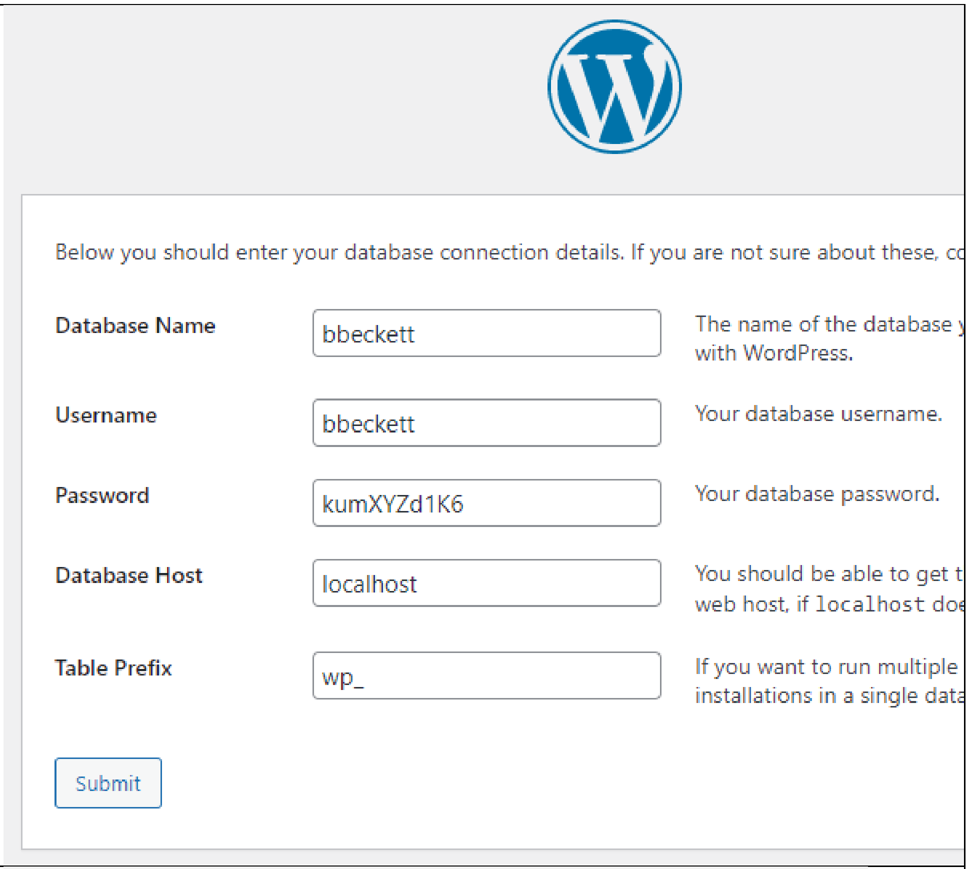
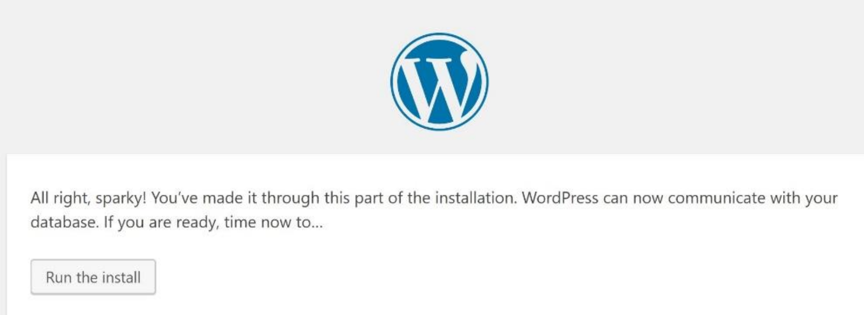
1. A set of Web pages in HTML and PHP. This is the engine that creates your site on-the-fly from information in the template and database. We'll be placing it in a directory within your account named "wordpress."
2. One or more templates. Templates contain designs for various types of Web sites. Templates are loaded onto the server in your WordPress area. When developing a Web site you may get a template from one of four places:
    a. WordPress comes with a few templates pre-installed, and you just select the one you want.
    b. There are numerous sites on the Web that provide freely-downloadable templates.
    c. There are sites on the Web that will sell you WordPress Templates.
    d. With a programming background, you can create a template either from one that already exists or from scratch.
3. A database. This contains information such as documents and your specific site's organization.

Note: Your Homework Server password and your WordPress password do not need to be the same. There is no reason they can't, however. So feel free to do it that way.

Let's install WordPress. As is the case with most hosting services, I've set up a Web page that does the installation.

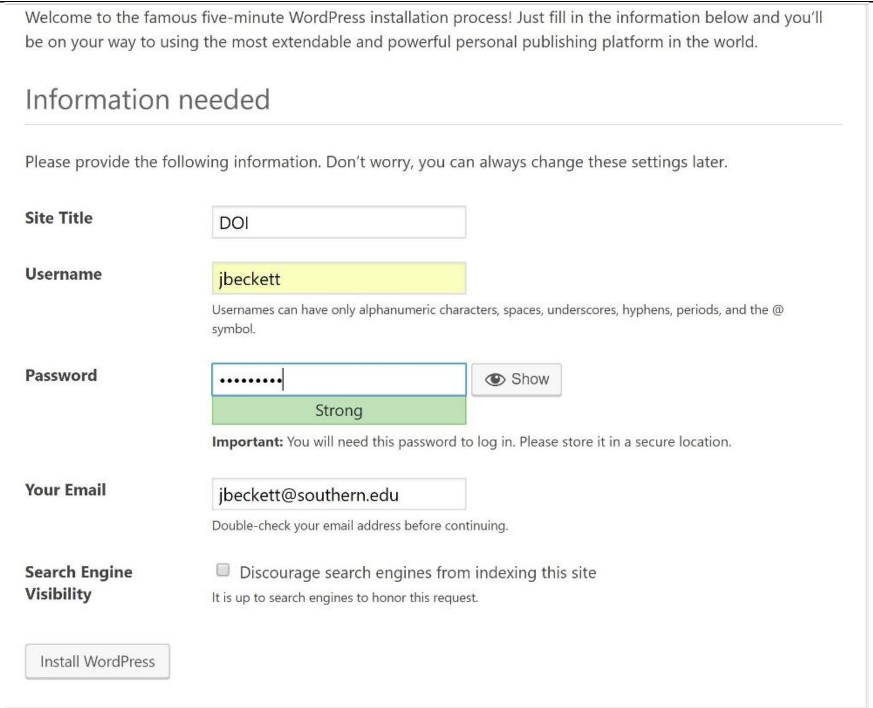
HW10: Use WordPress to create a Web site for the Diffusion of Innovations material we used in HW07.	
You'll start at https://hw.cs.southern.edu/wordpressreset.php Note this is an "https" address – not merely "http". This means the install process requires encryption. We do this because your email password is required. This is much like getting your Homework Server account. But you'll have to enter your WordPress administrative password again later.	 Word Press Request Form hw.cs.southern.edu/wpsetup/ eClass WX SOUTHERN BHP ARCHIVE HOME CPTE316 CP **Word Press setup** This site is used to prepare your Word Press site. Please enter the following information to p Username (don't include southern.edu): ______ @southern.edu ID number: ______
This page should show. Close your browser.	 hw.cs.southern.edu/wpsetup/wpsendmail.php eClass WX SOUTHERN BHP ARCHIVE HOME CPTE316 CPT **Preparing your Word Press site** Welcome, John Beckett! Watch your email for setup instructions. You may close this window
The email message will look something like this. Click on the link.	Your Word Press site is ready. You need to claim it within 5 minutes or redo your request.Go to https://hw.cs.southern.edu/wpsetup/activate.php?token=4469Q1Y3OTAO6VP3VD94 for the next step.
The response should look like this.	As requested, your WordPress site has been created (or re-created). Your task will be to add content and functionality. You will receive an email with further instructions when your site is set up. This should occur within two minutes. Check your spam folder.
You should receive an email like this. It may have gone in your spam folder. Save that database password – you'll need it later! Now click the link you were given.	Your WordPress account is created. Next you should log into [https://hw.cs.southern.edu/*myusername*/wp-admin](https://hw.cs.southern.edu/myusername/wp-admin) to set it up. Your dabatabase password for installation will be: kumXYZd1K6
At this point you're trying to access your Homework Server account. So like always, the system must have your login credentials.	 A username and password are being requested by https://hw.cs.southern.edu. The site says: "HomeWork Server" User Name: rbeckett Password: ••••••••

Select your language. WordPress assumes you want English (United States), so just click “Continue”.	 English (United States) Afrikaans العربية العربية المغربية অসমীয়া Azərbaycan dili گؤنئی آذربایجان Беларуская мова Български বাংলা བོད་ཡིག Bosanski Català Cebuano Čeština Cymraeg Dansk Deutsch (Sie) Deutsch (Schweiz) Deutsch Deutsch (Schweiz, Du) Continue
Things look good to go, so click “Let’s Go!”.	 Welcome to WordPress. Before getting started, we need some information on the database. You will need to know the following items before proceeding. 1. Database name 2. Database username 3. Database password 4. Database host 5. Table prefix (if you want to run more than one WordPress in a single database) We’re going to use this information to create a `wp-config.php` file. **If for any reason this automatic file creation doesn’t work, don’t worry. All this does is fill in the database information to a configuration file. You may also simply open `wp-config-sample.php` in a text editor, fill in your information, and save it as `wp-config.php`.** Need more help? We got it. In all likelihood, these items were supplied to you by your Web Host. If you don’t have this information, then you will need to contact them before you can continue. If you’re all ready…

You'll have to fill in some info. • Put your username and an underscore in front of "wordpress" for the database name • Your username is your username • Sorry, the password will show when you type it. I'm not showing mine! • Leave the host and table prefix alone.	 Below you should enter your database connection details. If you are not sure about these, co Database Name: bbeckett — The name of the database y with WordPress. Username: bbeckett — Your database username. Password: kumXYZd1K6 — Your database password. Database Host: localhost — You should be able to get t web host, if `localhost` doe Table Prefix: wp_ — If you want to run multiple installations in a single data Submit
Ready to go…	 All right, sparky! You've made it through this part of the installation. WordPress can now communicate with your database. If you are ready, time now to… Run the install

This time you have the option of unclicking "show" before typing in your password.

When you click "Install WordPress" it takes 15-30 seconds, so don't expect it to be instant.



Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Don't worry, you can always change these settings later.

Site Title DOI

Username jbeckett

Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.

Password •••••••••

Strong

Important: You will need this password to log in. Please store it in a secure location.

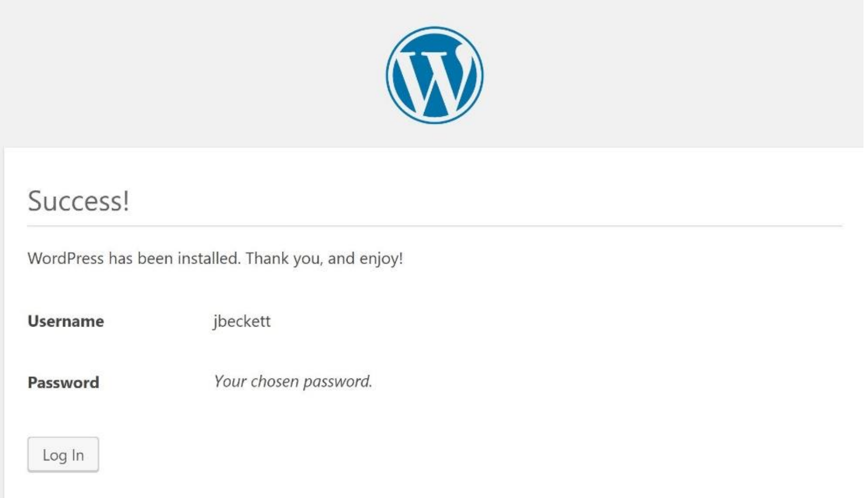
Your Email jbeckett@southern.edu

Double-check your email address before continuing.

Search Engine Visibility □ Discourage search engines from indexing this site

It is up to search engines to honor this request.

The basic install of WordPress is complete. You can click the "Log In" button now, or use the administrative link you got from the email, to continue.



Success!

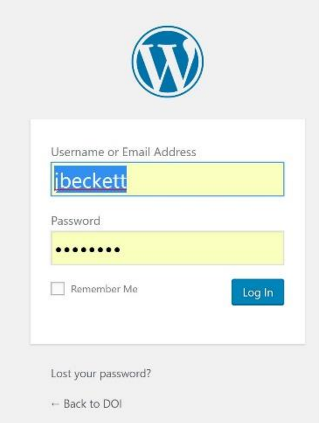
WordPress has been installed. Thank you, and enjoy!

Username jbeckett

Password *Your chosen password.*

Time to log into the administrative (back-end) interface of your WordPress site.

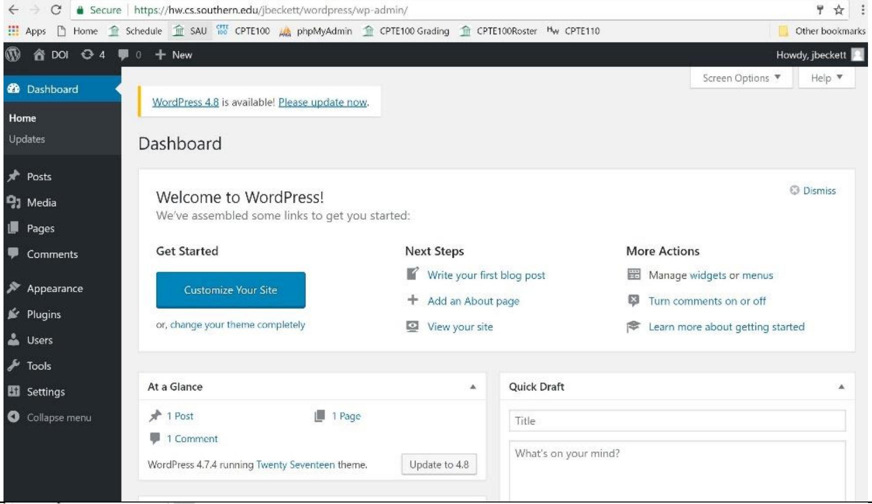
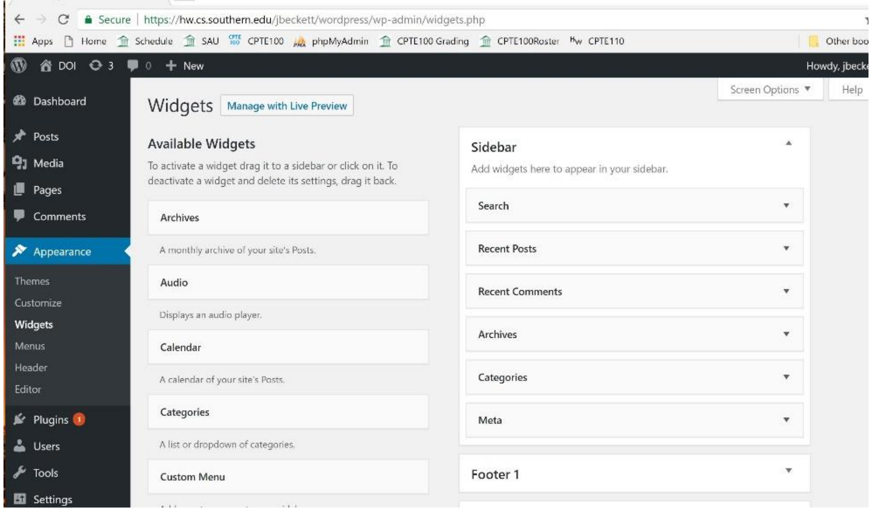
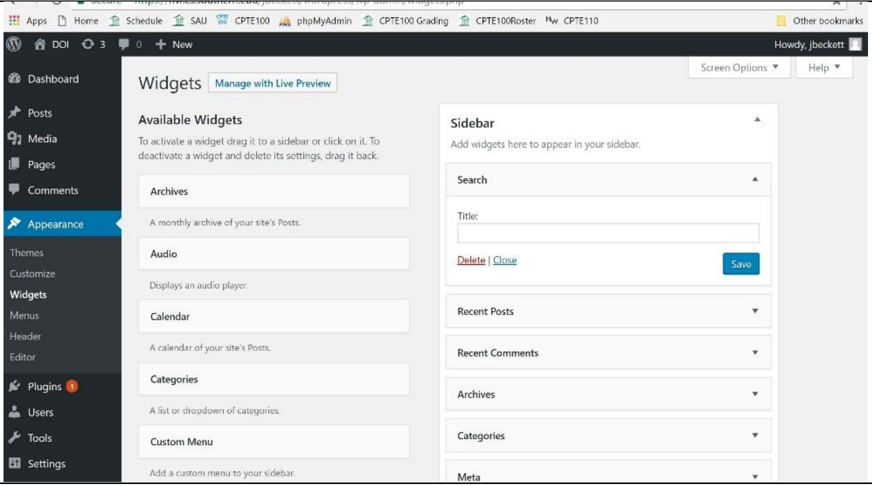
Enter your username and password to log in. This is the password you used when creating your WordPress site.



Username or Email Address jbeckett

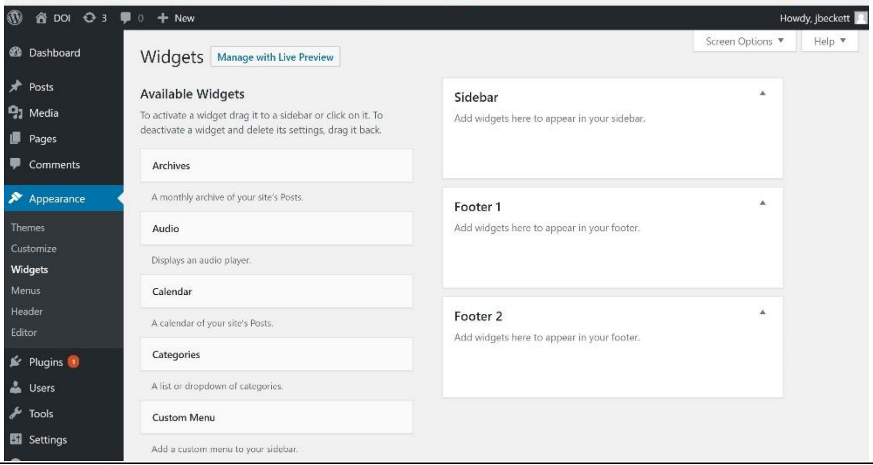
Password ••••••••

□ Remember Me

See that note that a new version is available? Go ahead and update. When it's done, click "Dashboard" to continue.	 WordPress 4.8 is available! Please update now. Welcome to WordPress! We've assembled some links to get you started: WordPress 4.7.4 running Twenty Seventeen theme. https://hw.cs.southern.edu/jbeckett/wordpress/wp-admin/ Dashboard Home Updates Posts Media Pages Comments Appearance Plugins Users Tools Settings Collapse menu Howdy, jbeckett Screen Options Help Dismiss Get Started Customize Your Site or, change your theme completely Next Steps Write your first blog post Add an About page View your site More Actions Manage widgets or menus Turn comments on or off Learn more about getting started At a Glance 1 Post 1 Page 1 Comment Update to 4.8 Quick Draft Title What's on your mind?
Under "More Actions" click "Manage widgets."	 To activate a widget drag it to a sidebar or click on it. To deactivate a widget and delete its settings, drag it back. Add widgets here to appear in your sidebar. https://hw.cs.southern.edu/jbeckett/wordpress/wp-admin/widgets.php Widgets Manage with Live Preview Available Widgets Archives A monthly archive of your site's Posts. Audio Displays an audio player. Calendar A calendar of your site's Posts. Categories A list or dropdown of categories. Custom Menu Sidebar Search Recent Posts Recent Comments Archives Categories Meta Footer 1 Dashboard Posts Media Pages Comments Appearance Themes Customize Widgets Menus Header Editor Plugins Users Tools Settings
Here we've clicked on the search widget. We can then delete it. We need to do this with all the sidebar widgets.	 To activate a widget drag it to a sidebar or click on it. To deactivate a widget and delete its settings, drag it back. Add widgets here to appear in your sidebar. Widgets Manage with Live Preview Available Widgets Archives Audio Calendar Categories Custom Menu Add a custom menu to your sidebar. Sidebar Search Title: Delete | Close Save Recent Posts Recent Comments Archives Categories Meta

You've deleted all the widgets when you see these three "Add widgets" blocks.

Now let's change the theme. Return to the dashboard and click "Appearance" .. "Themes"



To activate a widget drag it to a sidebar or click on it. To deactivate a widget and delete its settings, drag it back.

A monthly archive of your site's Posts.

Displays an audio player.

A calendar of your site's Posts.

A list or dropdown of categories.

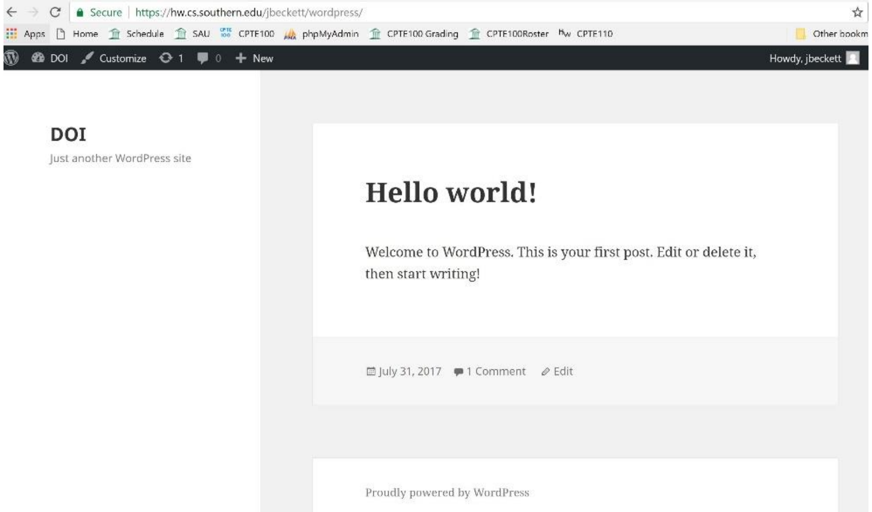
Add a custom menu to your sidebar.

Add widgets here to appear in your sidebar.

Add widgets here to appear in your footer.

Add widgets here to appear in your footer.

A few themes appear. We'll be using "Twenty-Fifteen". Hover over it and click "Activate." Oops, it wants to be updated. Go ahead and "Update now" and click "DOI along the top to see how it's going.



Hello world!

Welcome to WordPress. This is your first post. Edit or delete it, then start writing!

Proudly powered by WordPress

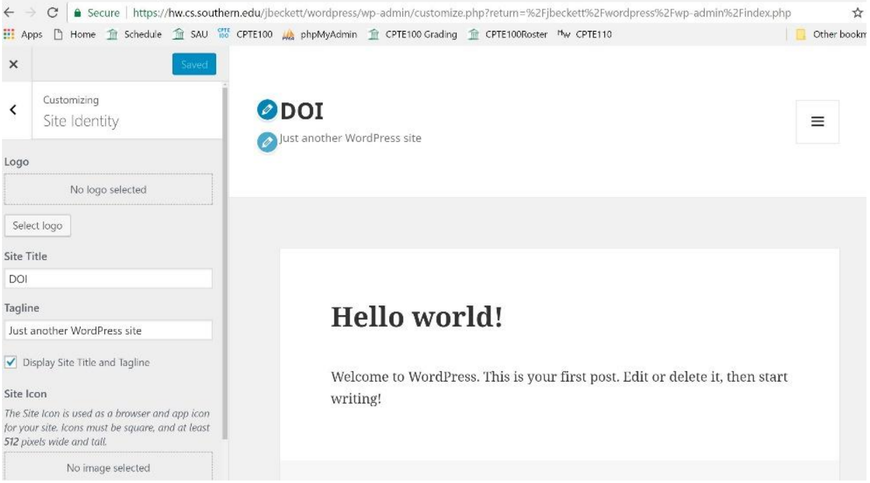
Now click the WordPress icon on the left of the top line and click "Dashboard"

Then select "Appearance".."Site Identity"

This offers us the chance to change the logo, title, and tagline. Un-click "Display site title and tagline"

Then install your logo by clicking "Select logo" and getting it from the former site.

You'll need to adjust the cropping after you select it.


Secure
https://hw.cs.southern.edu/jbeckett/wordpress/wp-admin/customize.php?return=%2Fjbeckett%2Fwordpress%2Fwp-admin%2Findex.php
Apps
Home
Schedule
SAU
CPTE100
phpMyAdmin
CPTE100 Grading
CPTE100Roster
CPTE110
Other bookm
×
Saved
Customizing
Site Identity
Select logo
DOI
Just another WordPress site
≡

Logo: No logo selected

Site Title: DOI

Tagline: Just another WordPress site

☑ Display Site Title and Tagline

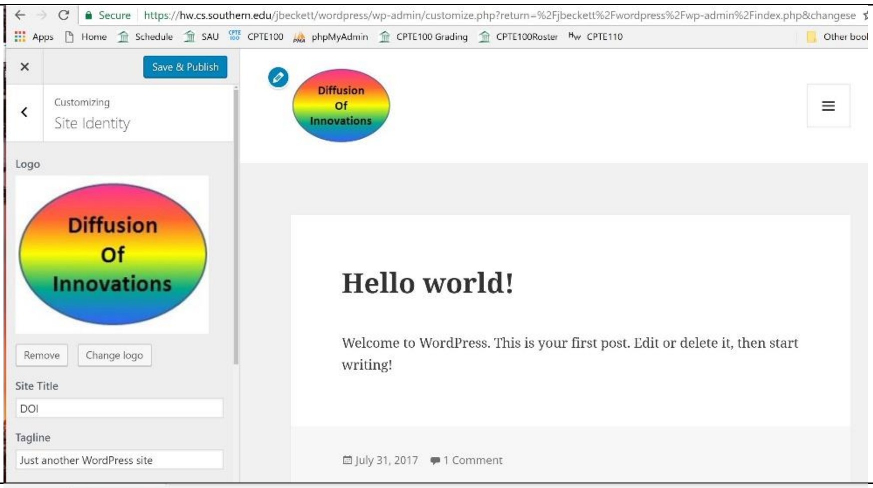
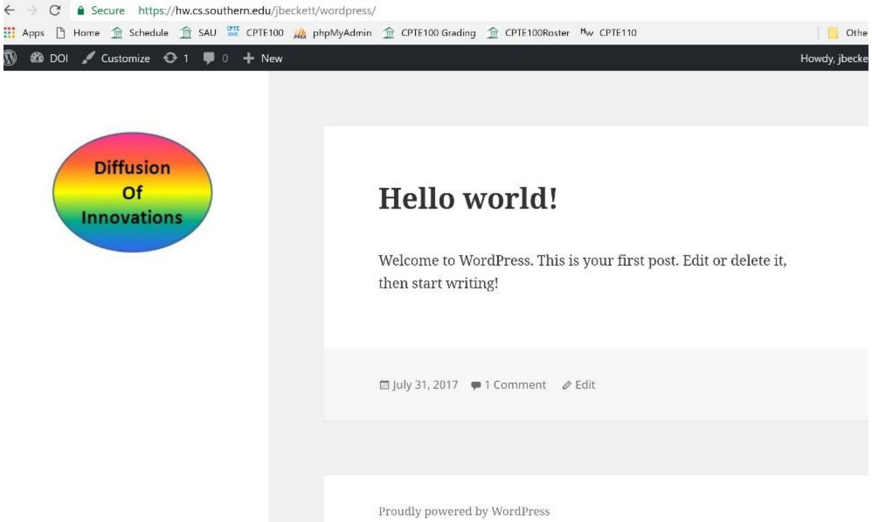
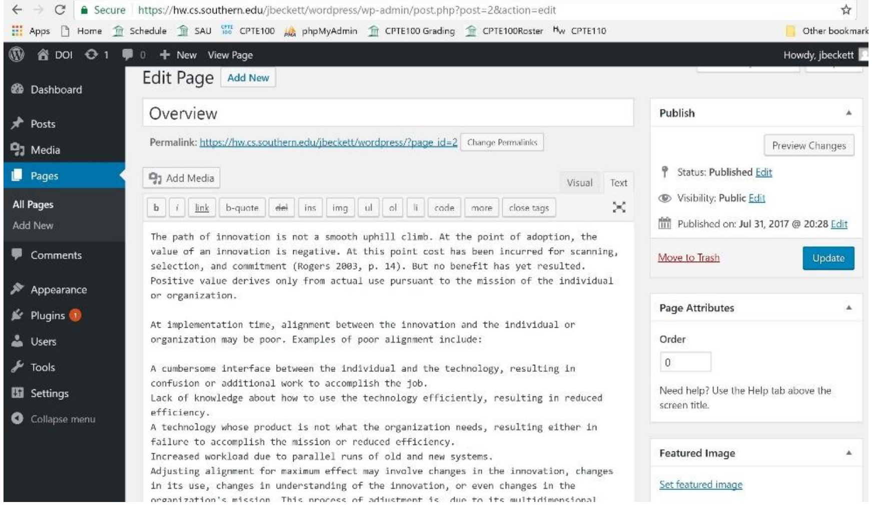
Site Icon

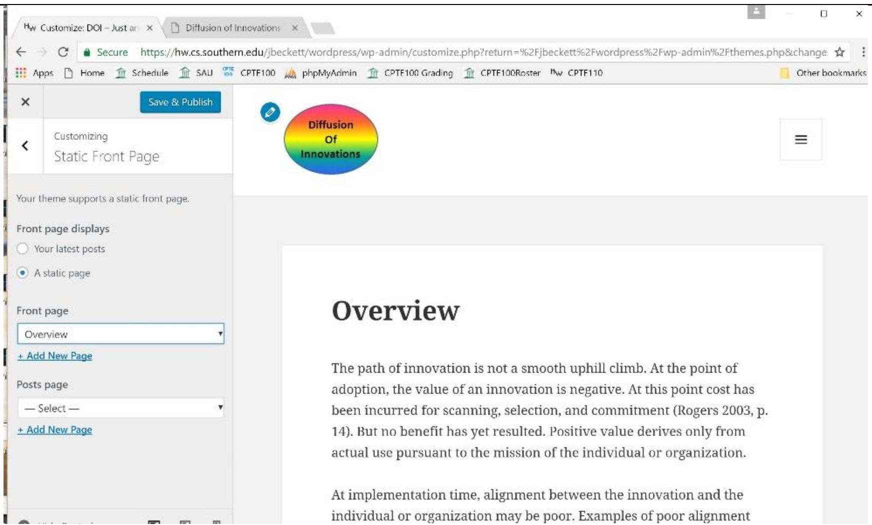
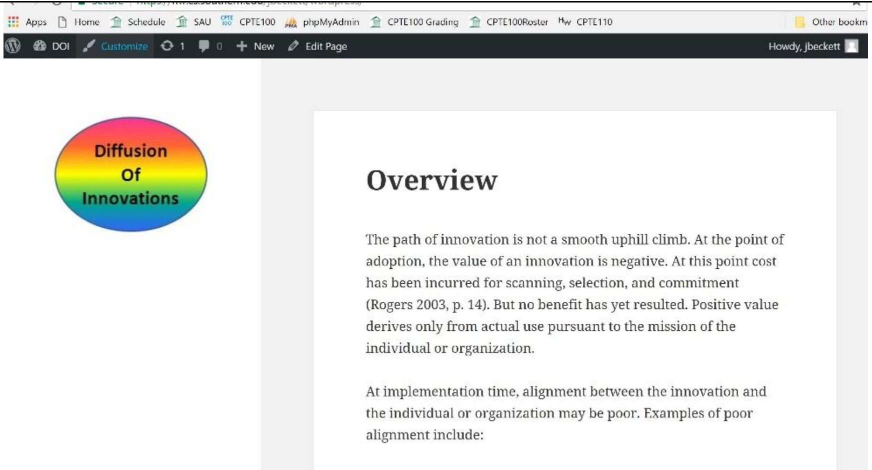
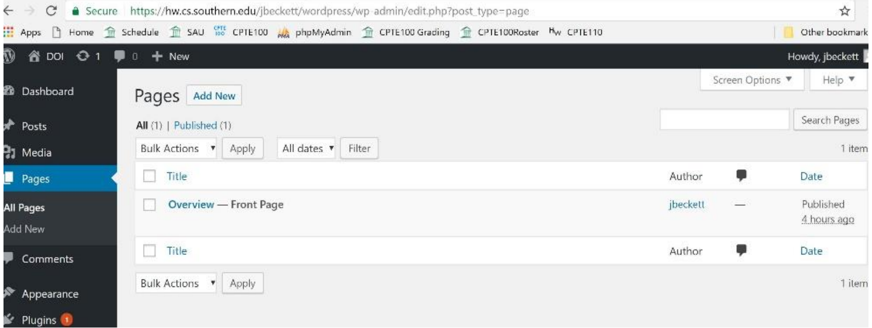
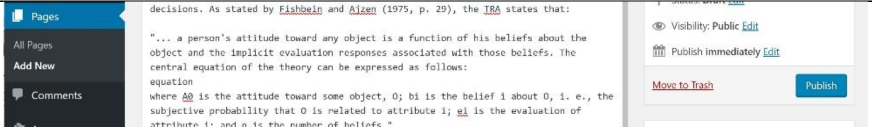
The Site Icon is used as a browser and app icon for your site. Icons must be square, and at least 512 pixels wide and tall.

No image selected

Hello world!

Welcome to WordPress. This is your first post. Edit or delete it, then start writing!

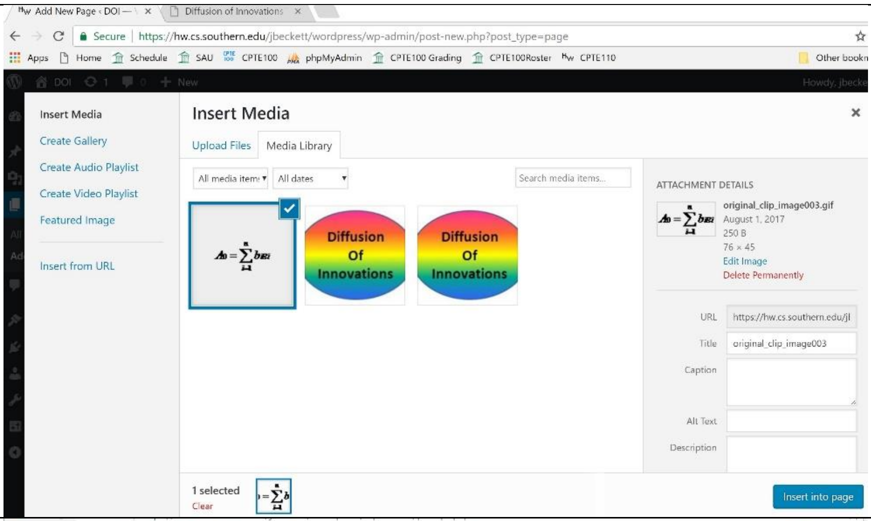
"Save and Publish" – then the "X" on the left, returning you to the dashboard. Let's take another look at our site now by clicking "DOI" along the top.	 Site Title: DOI Tagline: Just another WordPress site Hello world! Welcome to WordPress. This is your first post. Edit or delete it, then start writing! July 31, 2017 1 Comment
OK, the logo is in place but we don't see a menu or content.	 Hello world! Welcome to WordPress. This is your first post. Edit or delete it, then start writing! July 31, 2017 1 Comment Edit Proudly powered by WordPress
Go back to the Dashboard and select Pages. Hover over the sample page and select the edit function. You'll be replacing the heading and content. For the heading on this page, use "Overview" and for the content, grab the content from the former site's index page. Once you have it correct, click "Update"	 Edit Page Overview Permalink: https://hw.cs.southern.edu/jbeckett/wordpress/?page_id=2 The path of innovation is not a smooth uphill climb. At the point of adoption, the value of an innovation is negative. At this point cost has been incurred for scanning, selection, and commitment (Rogers 2003, p. 14). But no benefit has yet resulted. Positive value derives only from actual use pursuant to the mission of the individual or organization. At implementation time, alignment between the innovation and the individual or organization may be poor. Examples of poor alignment include: A cumbersome interface between the individual and the technology, resulting in confusion or additional work to accomplish the job. Lack of knowledge about how to use the technology efficiently, resulting in reduced efficiency. A technology whose product is not what the organization needs, resulting either in failure to accomplish the mission or reduced efficiency. Increased workload due to parallel runs of old and new systems. Adjusting alignment for maximum effect may involve changes in the innovation, changes in its use, changes in understanding of the innovation, or even changes in the organization's mission. This process of adjustment is, due to its multidimensional Status: Published Visibility: Public Published on: Jul 31, 2017 @ 20:28 Order: 0 Need help? Use the Help tab above the screen title.

Now go to "Appearance" .. "Customize" and select "Static Front Page" – then select the "A static page" option. For the front page, select the overview page you've just created. The result should look like this. Click "Save & Publish" and "X"-it from this screen.	 https://hw.cs.southern.edu/jbeckett/wordpress/wp-admin/customize.php?return=%2Fjbeckett%2Fwordpress%2Fwp-admin%2Fthemes.php&change Save & Publish Customizing Static Front Page Your theme supports a static front page. Front page displays ○ Your latest posts ● A static page Front page: Overview + Add New Page Posts page: — Select — + Add New Page Diffusion Of Innovations Overview The path of innovation is not a smooth uphill climb. At the point of adoption, the value of an innovation is negative. At this point cost has been incurred for scanning, selection, and commitment (Rogers 2003, p. 14). But no benefit has yet resulted. Positive value derives only from actual use pursuant to the mission of the individual or organization. At implementation time, alignment between the innovation and the individual or organization may be poor. Examples of poor alignment
And…here is what the site view looks like now. The front page is all there, but no menu..yet. We need to add pages first.	 DOI Customize 1 0 + New Edit Page Howdy, jbeckett Diffusion Of Innovations Overview The path of innovation is not a smooth uphill climb. At the point of adoption, the value of an innovation is negative. At this point cost has been incurred for scanning, selection, and commitment (Rogers 2003, p. 14). But no benefit has yet resulted. Positive value derives only from actual use pursuant to the mission of the individual or organization. At implementation time, alignment between the innovation and the individual or organization may be poor. Examples of poor alignment include:
Back at the "Pages" page, we see that we have only one page. Click "Add New."	 https://hw.cs.southern.edu/jbeckett/wordpress/wp-admin/edit.php?post_type=page Pages Add New All (1) | Published (1) Bulk Actions Apply All dates Filter 1 item □ Title — Author — Date □ Overview — Front Page — jbeckett — — — Published 4 hours ago □ Title — Author — Date Bulk Actions Apply 1 item
The second page is "Original Models" – and we get our content from the Former Site. Oops – that equation didn't get in there, and was replaced by its alt= information.	 decisions. As stated by Fishbein and Ajzen (1975, p. 29), the TRA states that: "... a person's attitude toward any object is a function of his beliefs about the object and the implicit evaluation responses associated with those beliefs. The central equation of the theory can be expressed as follows: equation where A0 is the attitude toward some object, O; bi is the belief i about O, i. e., the subjective probability that O is related to attribute i; ei is the evaluation of attribute i; and n is the number of beliefs." Visibility: Public Edit Publish immediately Edit Move to Trash Publish

Highlight that word "equation" and click "Add Media."

You'll need to "Upload Files" that equation graphic from the Former Site, then "Insert into page"

When you get back to the page-entry page, click "Publish"



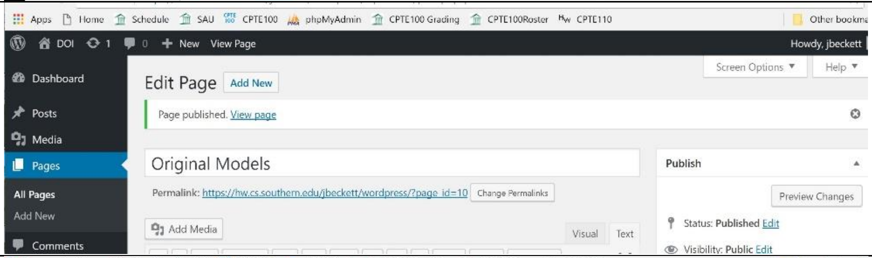
Insert Media

ATTACHMENT DETAILS

original_clip_image003.gif
August 1, 2017
250 B
76 × 45
Edit Image
Delete Permanently

URL: https://hw.cs.southern.edu/jl
Title: original_clip_image003
Caption:
Alt Text:
Description:

Now you can click "View page" to see how that worked.



Edit Page

Page published. View page

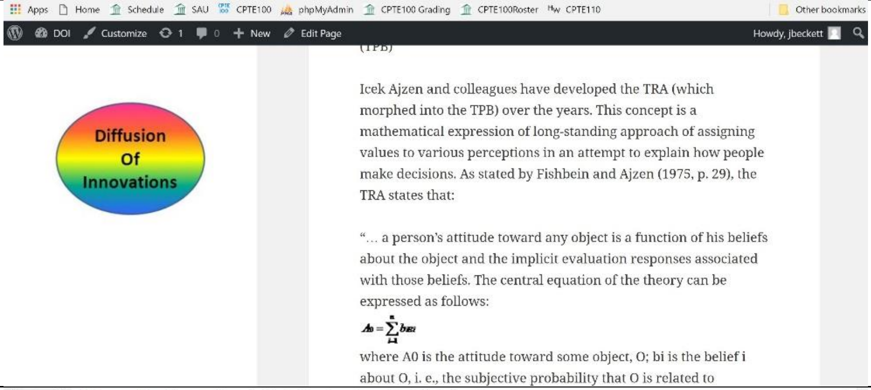
Original Models

Permalink: https://hw.cs.southern.edu/jbeckett/wordpress/?page_id=10

Status: Published Edit

Visibility: Public Edit

Scrolling down, we see that it did!


Apps Home Schedule SAU CPTE100 phpMyAdmin CPTE100 Grading CPTE100Roster CPTE110
Other bookmarks
DOI Customize 1 0 New Edit Page
Howdy, jbeckett
Diffusion Of Innovations

(TPB)

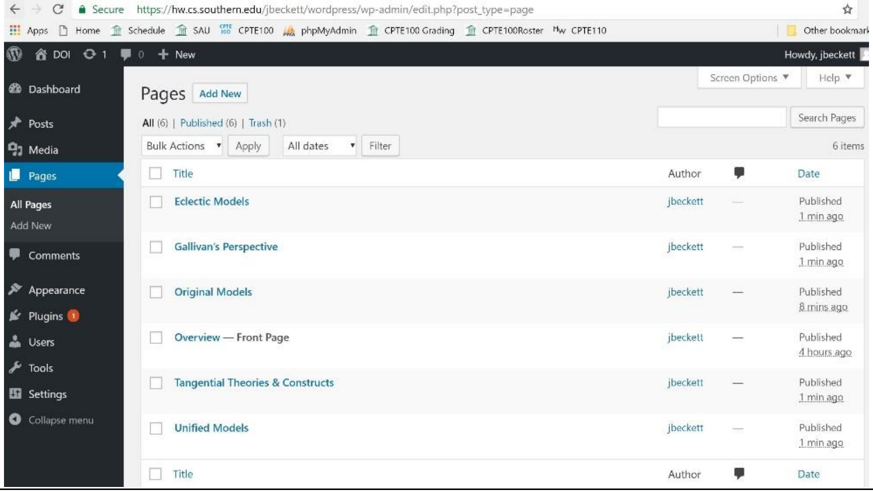
Icek Ajzen and colleagues have developed the TRA (which morphed into the TPB) over the years. This concept is a mathematical expression of long-standing approach of assigning values to various perceptions in an attempt to explain how people make decisions. As stated by Fishbein and Ajzen (1975, p. 29), the TRA states that:

"… a person's attitude toward any object is a function of his beliefs about the object and the implicit evaluation responses associated with those beliefs. The central equation of the theory can be expressed as follows:

$A_0 = \sum b_i e_i$

where A0 is the attitude toward some object, O; bi is the belief i about O, i. e., the subjective probability that O is related to

The other pages are done the same way, except that they don't have graphics in them. After you're done your "Pages" page should look like this:



Pages

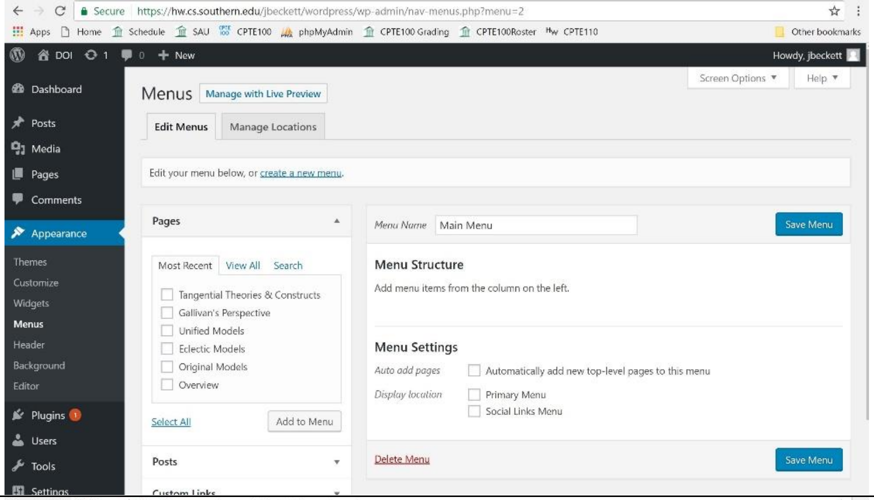
All (6) | Published (6) | Trash (1)

6 items

Title	Author		Date
Eclectic Models	jbeckett	—	Published 1 min ago
Gallivan's Perspective	jbeckett	—	Published 1 min ago
Original Models	jbeckett	—	Published 8 mins ago
Overview — Front Page	jbeckett	—	Published 4 hours ago
Tangential Theories & Constructs	jbeckett	—	Published 1 min ago
Unified Models	jbeckett	—	Published 1 min ago
Title	Author		Date

These pages, however, aren't visible on the site. For that we'll need a menu.

Back to the dashboard, "Appearance" .. "Menus" – then create a menu, calling it "Main Menu" for lack of a better name.


Secure | https://hw.cs.southern.edu/jbeckett/wordpress/wp-admin/nav-menus.php?menu=2
Apps Home Schedule SAU CPTE100 phpMyAdmin CPTE100 Grading CPTE100Roster CPTE110 Other bookmarks
DOI 1 0 New Howdy, jbeckett
Dashboard
Posts
Media
Pages
Comments
Appearance
Themes
Customize
Widgets
Menus
Header
Background
Editor
Plugins 1
Users
Tools
Settings
Screen Options ▾ Help ▾
Menus Manage with Live Preview
Edit Menus Manage Locations
Pages ▴
Most Recent View All Search
Select All Add to Menu
Posts ▾
Custom Links ▾
Save Menu
Delete Menu
Save Menu

Edit your menu below, or create a new menu.

- □ Tangential Theories & Constructs
- □ Gallivan's Perspective
- □ Unified Models
- □ Eclectic Models
- □ Original Models
- □ Overview

Menu Name Main Menu

Menu Structure

Add menu items from the column on the left.

Menu Settings

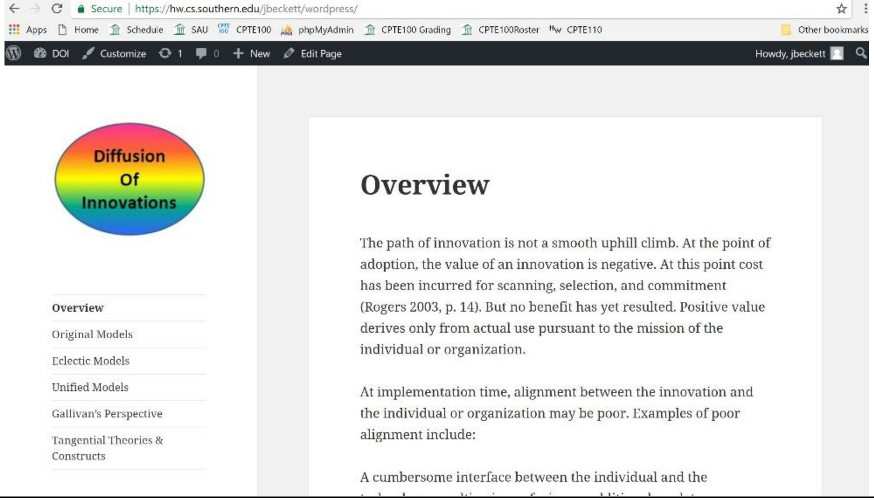
Auto add pages □ Automatically add new top-level pages to this menu

Display location □ Primary Menu
□ Social Links Menu

On the left you will see the various pages you've created. Click all of them, and click "Add to Menu."

Next, drag them into the correct order.

Now on the right click "Primary Menu" – then click "Save Menu." Now we have a working site:


Secure | https://hw.cs.southern.edu/jbeckett/wordpress/
Apps Home Schedule SAU CPTE100 phpMyAdmin CPTE100 Grading CPTE100Roster CPTE110 Other bookmarks
DOI Customize 1 0 New Edit Page Howdy, jbeckett
Diffusion
Of
Innovations
Overview
Original Models
Eclectic Models
Unified Models
Gallivan's Perspective
Tangential Theories & Constructs

Overview

The path of innovation is not a smooth uphill climb. At the point of adoption, the value of an innovation is negative. At this point cost has been incurred for scanning, selection, and commitment (Rogers 2003, p. 14). But no benefit has yet resulted. Positive value derives only from actual use pursuant to the mission of the individual or organization.

At implementation time, alignment between the innovation and the individual or organization may be poor. Examples of poor alignment include:

A cumbersome interface between the individual and the

Lesson 11 – Using Content Management Systems - 2

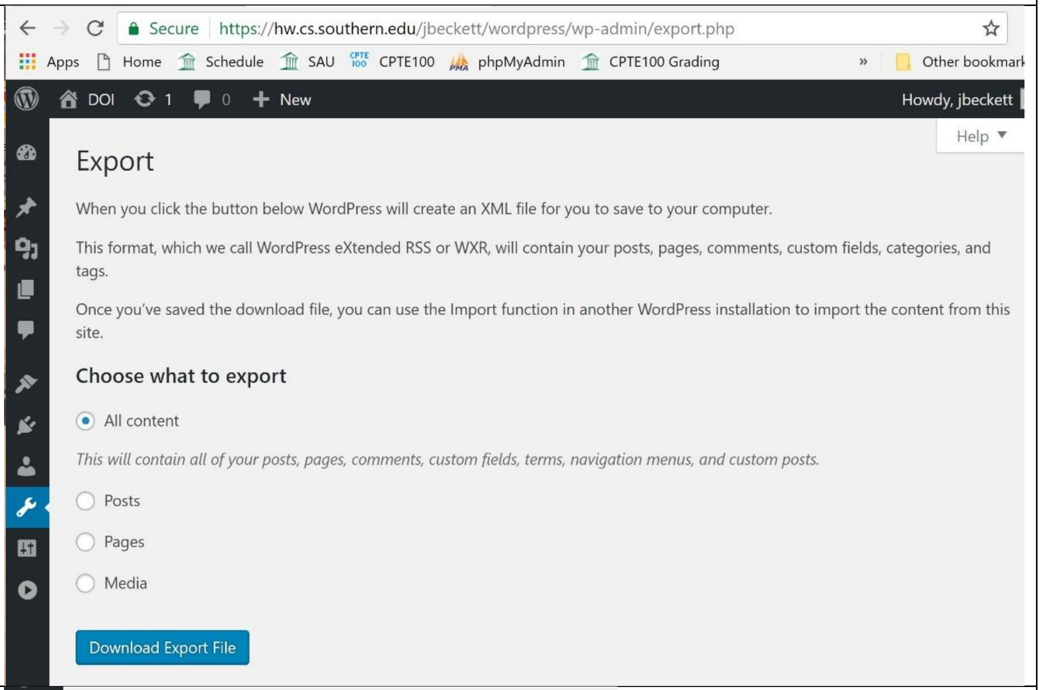
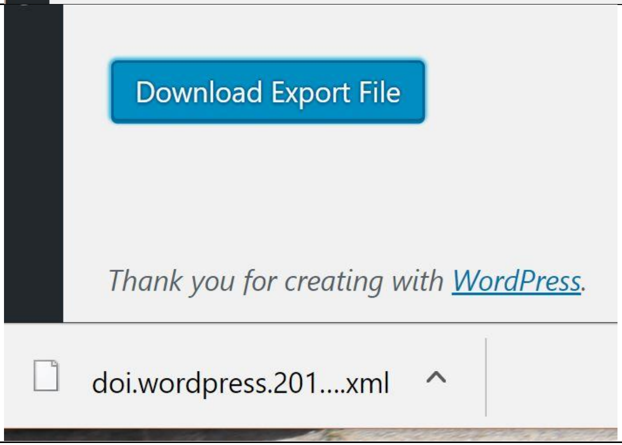
In our last lesson we got a taste of how to present information using the WordPress content management system. As we went on this tour, you probably noticed that there is a *lot* more that one can do. WordPress is a platform that a group of people can use, each with their own set of privileges in terms of what they can see and what they can create. If you can live with the look and feel of WordPress, you can get on with whatever drove you to using the Web instead of spending your life programming HTML and CSS.

But there's that gotcha: "if you can live with it." The look and feel of a WordPress site is dictated by something called a template. A template is a group of files, most of them CSS, defining all sorts of details. To reiterate what we mentioned in the previous lesson, templates can come from several places:

1. WordPress Comes with a few templates pre-installed, and you just select the one you want.
2. There are numerous sites on the Web that provide freely-downloadable templates.
3. There are sites on the Web that will sell you WordPress Templates.
4. With a programming background, you can create a template either from one that already exists or from scratch.

HW11: You will be doing several common tasks related to a Content Management System.

1. Make a backup by exporting the SQL from the database. Build the backup file in SQL format, then upload it to the HW11 directory of your Homework Server account.
2. Obtain a different template, install it, and configure it for our needs

Making a backup	
Log into the administrative interface and go to the dashboard. Select "Tools" .. "Export"	 Secure https://hw.cs.southern.edu/jbeckett/wordpress/wp-admin/export.php Apps Home Schedule SAU CPTE100 phpMyAdmin CPTE100 Grading » Other bookmark DOI 1 0 New Howdy, jbeckett Help ▼ Export When you click the button below WordPress will create an XML file for you to save to your computer. This format, which we call WordPress eXtended RSS or WXR, will contain your posts, pages, comments, custom fields, categories, and tags. Once you've saved the download file, you can use the Import function in another WordPress installation to import the content from this site. Choose what to export ● All content *This will contain all of your posts, pages, comments, custom fields, terms, navigation menus, and custom posts.* ○ Posts ○ Pages ○ Media
The backup file, which is in xml format, will be saved in your downloads folder. Upload to your HW11 folder on the Homework Server to turn it in.	 *Thank you for creating with WordPress.* doi.wordpress.201....xml

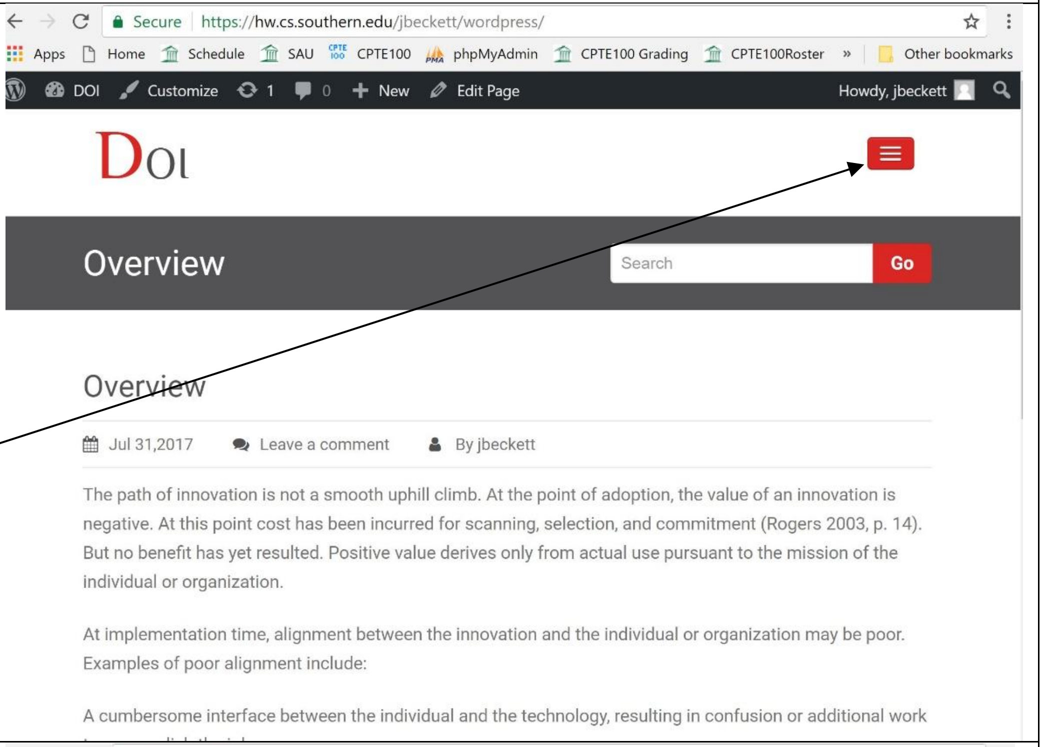
Installing a New Theme

In the Dashboard, go to "Appearance" … "Themes"

We want the "Rambo" theme. Click "Add New" and type "Rambo" into the search box.

Install it and activate it. Things look different now.

We see information about when a page was written, offer of a comment, and we lost our menu. Oh – it's that multi-bar thing on the right!



Overview

Jul 31,2017 Leave a comment By jbeckett

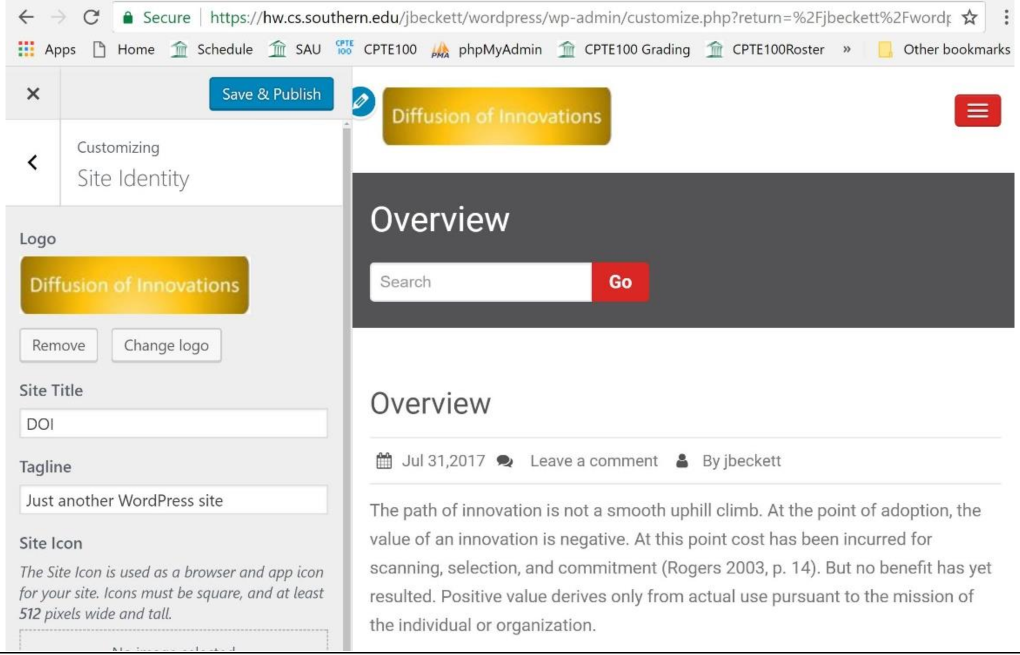
The path of innovation is not a smooth uphill climb. At the point of adoption, the value of an innovation is negative. At this point cost has been incurred for scanning, selection, and commitment (Rogers 2003, p. 14). But no benefit has yet resulted. Positive value derives only from actual use pursuant to the mission of the individual or organization.

At implementation time, alignment between the innovation and the individual or organization may be poor. Examples of poor alignment include:

A cumbersome interface between the individual and the technology, resulting in confusion or additional work

"Appearance" .. "Site Identity" .. "Select Logo" – re-install the logo. Alas, the logo cropping won't work. This is because the space allocated to a logo is smaller than our logo. So we'll use "DOInew.jpg"

Save and Publish.



Site Title: DOI

Tagline: Just another WordPress site

Site Icon

The Site Icon is used as a browser and app icon for your site. Icons must be square, and at least 512 pixels wide and tall.

Overview

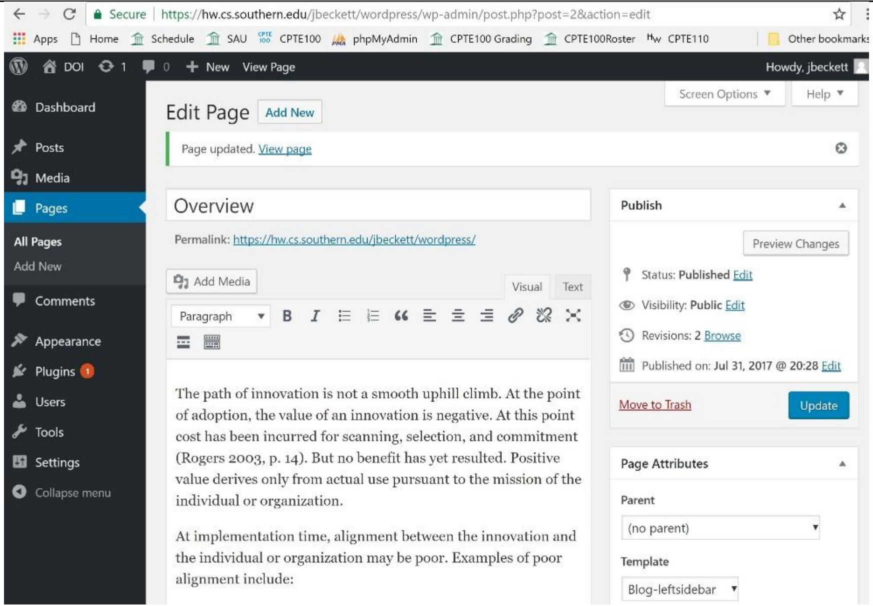
Jul 31,2017 Leave a comment By jbeckett

The path of innovation is not a smooth uphill climb. At the point of adoption, the value of an innovation is negative. At this point cost has been incurred for scanning, selection, and commitment (Rogers 2003, p. 14). But no benefit has yet resulted. Positive value derives only from actual use pursuant to the mission of the individual or organization.

Go into “Pages” and select the Overview, Edit it, selecting a Template of “Fullwidth.”

Click “Update”

Now we have a menu people can see!



Edit Page

Page updated. View page

Overview

Permalink: https://hw.cs.southern.edu/jbeckett/wordpress/

The path of innovation is not a smooth uphill climb. At the point of adoption, the value of an innovation is negative. At this point cost has been incurred for scanning, selection, and commitment (Rogers 2003, p. 14). But no benefit has yet resulted. Positive value derives only from actual use pursuant to the mission of the individual or organization.

At implementation time, alignment between the innovation and the individual or organization may be poor. Examples of poor alignment include:

Status: Published Edit

Visibility: Public Edit

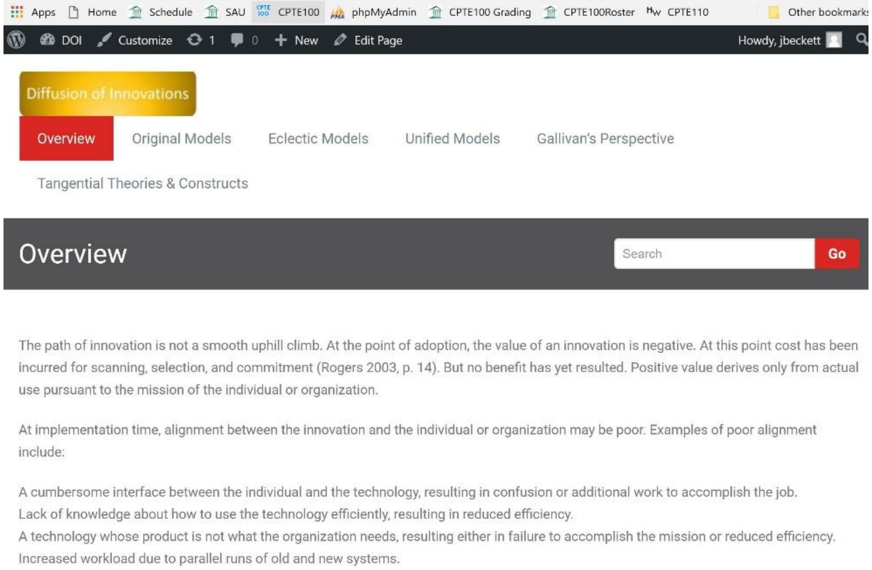
Revisions: 2 Browse

Published on: Jul 31, 2017 @ 20:28 Edit

Parent: (no parent)

Template: Blog-leftsidebar

Here is the result. Not particularly pretty, but it shows we can do something different and make it work.



Overview

The path of innovation is not a smooth uphill climb. At the point of adoption, the value of an innovation is negative. At this point cost has been incurred for scanning, selection, and commitment (Rogers 2003, p. 14). But no benefit has yet resulted. Positive value derives only from actual use pursuant to the mission of the individual or organization.

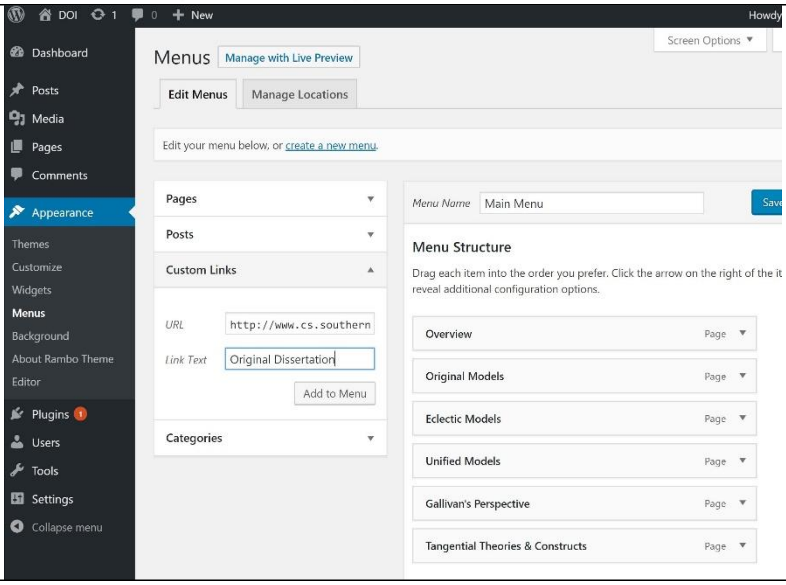
At implementation time, alignment between the innovation and the individual or organization may be poor. Examples of poor alignment include:

A cumbersome interface between the individual and the technology, resulting in confusion or additional work to accomplish the job.
Lack of knowledge about how to use the technology efficiently, resulting in reduced efficiency.
A technology whose product is not what the organization needs, resulting either in failure to accomplish the mission or reduced efficiency.
Increased workload due to parallel runs of old and new systems.

Now let’s add a link to the original dissertation.

Navigate to appearance, Menus, and add a custom link as shown.

The link is: http://www.cs.southern.edu/jbeckett/dissertation/



Menus

Edit your menu below, or create a new menu.

URL: http://www.cs.southern

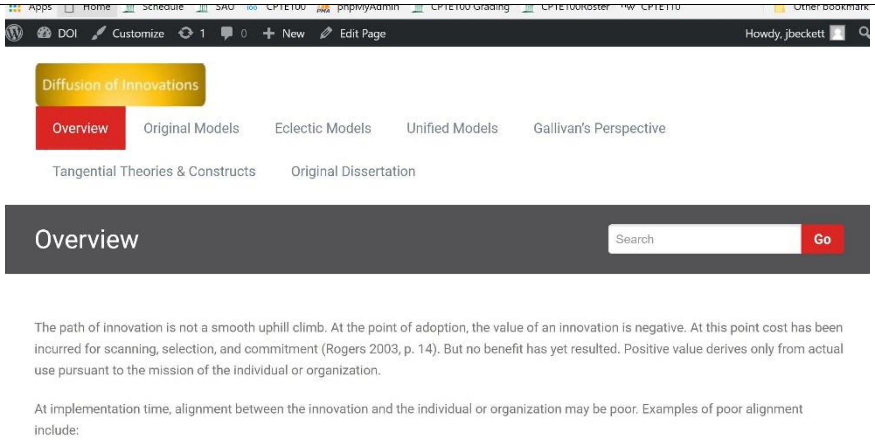
Link Text: Original Dissertation

Menu Name: Main Menu

Menu Structure

Drag each item into the order you prefer. Click the arrow on the right of the it reveal additional configuration options.

Overview — Page
Original Models — Page
Eclectic Models — Page
Unified Models — Page
Gallivan's Perspective — Page
Tangential Theories & Constructs — Page

As a result, we now have another link available!	 Apps Home Schedule SAU CPTE100 phpMyAdmin CPTE100 Grading CPTE100Roster CPTE110 Other bookmark DOI Customize 1 0 New Edit Page Howdy, jbeckett Diffusion of Innovations Overview Original Models Eclectic Models Unified Models Gallivan's Perspective Tangential Theories & Constructs Original Dissertation Search Go Overview The path of innovation is not a smooth uphill climb. At the point of adoption, the value of an innovation is negative. At this point cost has been incurred for scanning, selection, and commitment (Rogers 2003, p. 14). But no benefit has yet resulted. Positive value derives only from actual use pursuant to the mission of the individual or organization. At implementation time, alignment between the innovation and the individual or organization may be poor. Examples of poor alignment include:

Lesson 12 – DNS

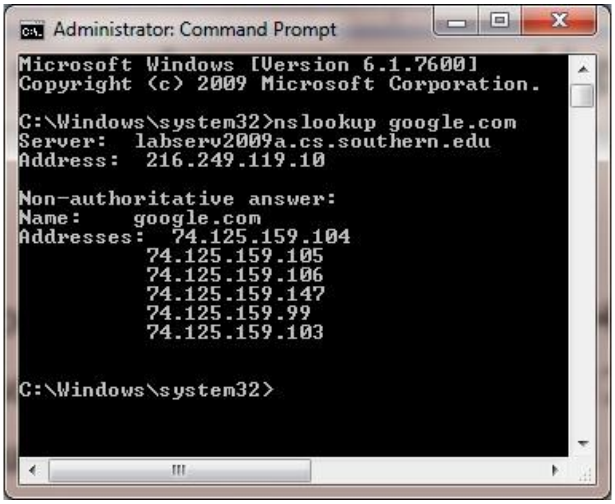
Find something you like to drink (I prefer ice water for this because it keeps me awake), because this isn't going to happen in a minute.

Where is your Web site? At the lowest level, one needs to use an **IP address**. It's sort of like latitude and longitude – but expressed in terms of how the Internet is organized. The geographical coordinates for the building where I work are 35˚02'45.67"N 85˚03'10.23"W. But to find my department on the Internet, you would go to 216.249.119.0. The Homework Server I use for this class is station 71. Internet people combine the address information, so the full address of the Homework Server as of August 2016 is 216.249.119.28. But don't use this number because it may change, such as when the server is replaced or upgraded.

> The scenario I'm describing here uses IPV4, the current version of IP addresses. It is composed of four numbers that each range from 0 through 255. If you're into math, you will know instantly that this number doesn't have enough variations to handle all the IP addresses we will need in the near future. Accordingly, this scheme is being replaced by IPV6 - one that has much longer numbers in it and uses colons instead of periods. All of which makes this lesson even more important.

But nobody wants to remember all those numbers. Would you like to type 74.125.159.104 when you want to use Google? It's worse than that. 74.125.159.104 might be busy (ya think?) so you might have to try 74.125.159.105, ..106, ..107, and several others. And Google people would be going nuts because the load on their servers would be very unbalanced.

The solution for this is the **Domain Name System**, usually just called "DNS." You probably think of DNS as something that has to do with an error message you got. But it's really one of the most important inventions on the Internet. DNS is a function that allows you to enter a rememberable/pronounceable name and tells your browser where to go for that function. We can manually interrogate that function using a tool called "nslookup" that's available in the command prompt (this works in command line functions of Windows, Mac, and Linux!) This is what I got for google.com. Let's interpret those terms:



```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.

C:\Windows\system32>nslookup google.com
Server:  labserv2009a.cs.southern.edu
Address:  216.249.119.10

Non-authoritative answer:
Name:    google.com
Addresses:  74.125.159.104
          74.125.159.105
          74.125.159.106
          74.125.159.147
          74.125.159.99
          74.125.159.103


C:\Windows\system32>
```

Non-authoritative answer: means that somebody in my area looked this up recently, and DNS isn't bothering to go all the way out to Google's Internet Service Provider to get the answer – it's remembered (cached) locally.

Name: That's what I looked for.

Addresses: This is a list of IP addresses for servers that answer to google.com. One of the functions of DNS is to issue them in "round robin" fashion to automatically distribute workload among servers that receive many requests.

So all we have to do to send our browser to Google is type "google.com" in the URL window.

Great, but how do we get DNS to know about our server? DNS is a database system. To get our domain name into DNS, we must **register** it. There are many registrars (companies that have the ability to put things in the DNS database) available, and they may charge whatever they like for their services. Competition keeps prices reasonable. If a registrar wants $100 per year to register your domain, they are way out of line. The price should be under $20.

The first task finding a domain name is, of course, finding one that isn't already used. On the Internet, you can't have two different entities using the same domain name because DNS can only send it to one host (or, as in the case of Google, group of hosts). This is why you see some domain names that aren't quite what you expect (try hobby-lobby.com and hobbylobby.com).

As one might expect for a function so central to modern life, there are numerous ways unscrupulous cheat or make trouble for Web site operators. Here are a few – and you may rest assured that many more exist or will be created.

> The simplest is **fraudulent names**. I once registered a domain of net96.org. Somebody who didn't like the organization I represented registered net-96.org, and placed articles on their site that complained about our organization. Many organizations now register several names that look like their official name, and route them all to their main server. Look at homedepotsucks.com to see why this is important.
>
> A variation on this method came in when **non-Roman characters** became available in the DNS a few years ago. Now you may see a link in an email that looks perfectly legitimate, but actually sends you to a server in an Eastern European country where your money might be transferred out of USA jurisdiction. Never click on a link in an email unless you know who sent you the email. It's much safer to type URLs in yourself.
>
> A popular **scam** in the Internet business is sending forms to domain registrants up for renewal. The form looks like an very official business invoice. If signed and paid (the price is usually much higher than what you paid earlier), it authorizes an unethical registrar to take control of your domain. Surprise! Their prices are a lot more than what you've been paying. So once you have a domain registered for yourself or your organization, don't pay anybody except your original registrar for a renewal!

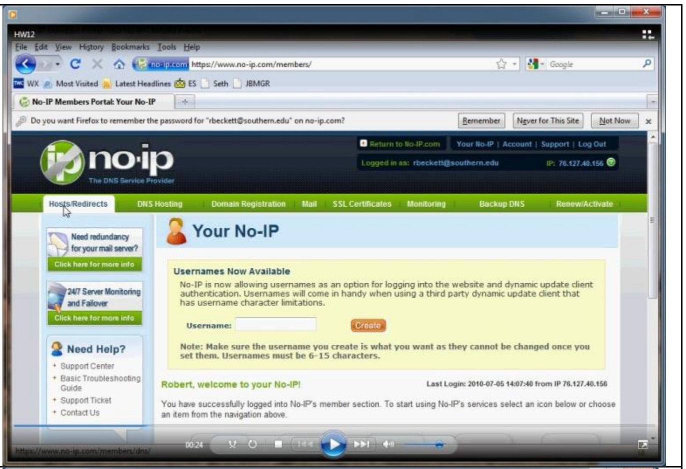
Having registered your domain name, you need to have it connect to your server's IP address. That's part of what your registrar will do for you. Each registrar's site is different. For the assignment in this lesson I'll have you register with No-IP.Com because they have a nice interface, are fairly reasonably priced (hint: Go Daddy is probably cheaper but perhaps harder to navigate), and will let you register a domain free. If you are accustomed to another free registrar, be my guest – I don't work for No-IP.COM.

HW12: This assignment has two parts:
1. Select and register a domain name, and link it to your WordPress Web site.
2. Place a simple Web page in your HW12 directory on the homework server with a link to this domain name.

Here's how I did it for Robert Beckett. Please note that when you are trying to get a free function from the Internet, the site may very well have changed. So read carefully.

Go to No-ip.com.

- Create an account
- Find a free hostname I can use. These are all sub-domains (having three parts instead of two) owned by no-ip.com.
- Click "Hosts/Redirects".


HW12
File Edit View History Bookmarks Tools Help
no-ip.com https://www.no-ip.com/members/
Google
WX Most Visited Latest Headlines ES Seth JBMGR
No-IP Members Portal: Your No-IP
Remember
Never for This Site
Not Now
no·ip
The DNS Service Provider
Return to No-IP.com
Your No-IP | Account | Support | Log Out
Hosts/Redirects
DNS Hosting
Domain Registration
Mail
SSL Certificates
Monitoring
Backup DNS
Renew/Activate
Need redundancy for your mail server?
Click here for more info
24/7 Server Monitoring and Failover
Click here for more info
Need Help?
Support Center
Basic Troubleshooting Guide
Support Ticket
Contact Us
Create
00:24

Do you want Firefox to remember the password for "rbeckett@southern.edu" on no-ip.com?

Logged in as: rbeckett@southern.edu IP: 76.127.40.156

Your No-IP

Usernames Now Available

No-IP is now allowing usernames as an option for logging into the website and dynamic update client authentication. Usernames will come in handy when using a third party dynamic update client that has username character limitations.

Username:

Note: Make sure the username you create is what you want as they cannot be changed once you set them. Usernames must be 6-15 characters.

Robert, welcome to your No-IP!

Last Login: 2010-07-05 14:07:40 from IP 76.127.40.156

You have successfully logged into No-IP's member section. To start using No-IP's services select an icon below or choose an item from the navigation above.

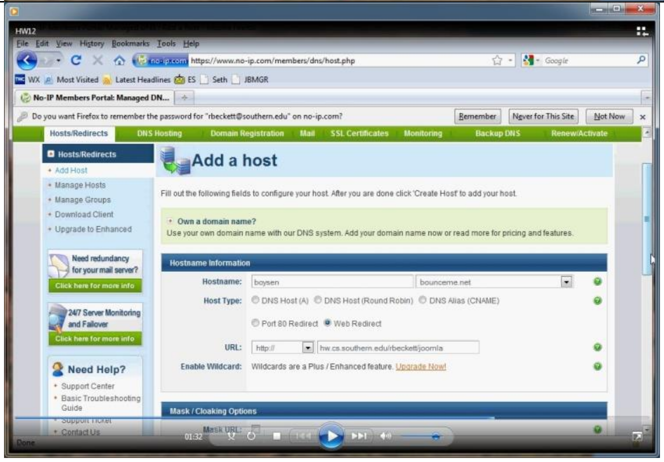
https://www.no-ip.com/members/dns/

- Select "Web Direct" on the host configuration.
- Enter the URL of the actual Website. In this case I used "hw.cs.southern.edu/rbeckett/wordpress".
- Now wait 10 minutes or so before attempting to use the new domain name.
  - If you don't wait while the underlying functions happen, your browser may become confused.

If your browser does become confused because you tried too soon or made an error and corrected it, exiting the browser and rebooting your computer might help.

(In the illustration shown I used joomla instead of WordPress – but you can make that substitution!)

In any case, test the URL from a different workstation.



Do you want Firefox to remember the password for "rbeckett@southern.edu" on no-ip.com?

Add a host

Fill out the following fields to configure your host. After you are done click 'Create Host' to add your host.

Own a domain name?
Use your own domain name with our DNS system. Add your domain name now or read more for pricing and features.

Hostname Information

Hostname: boysen bounceme.net

Host Type: ○ DNS Host (A) ○ DNS Host (Round Robin) ○ DNS Alias (CNAME)
○ Port 80 Redirect ● Web Redirect

URL: http:// hw.cs.southern.edu/rbeckett/joomla

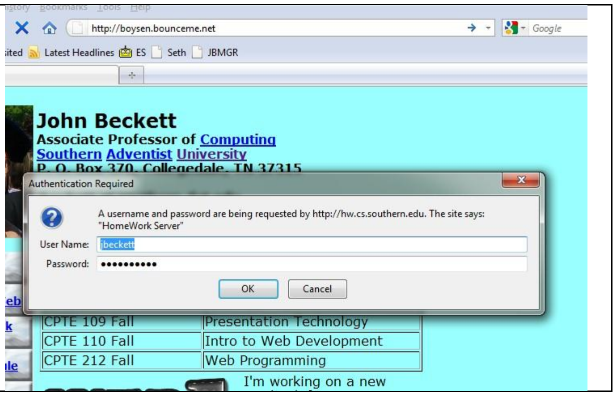
Enable Wildcard: Wildcards are a Plus / Enhanced feature.

Mask / Cloaking Options

Mask URL: □

Now for that simple Web page. The code I used is on the left, and the result when the link is clicked is on the right. It's asking me for my homework server username and password.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>HW12 - Using my own URL</title>
</head>
<body>
<p>To go to my Web site using my special DNS name,
click
<a href="http://boysen.bounceme.net">here</a>.</p>
</body>
</html>
```


istory Bookmarks Tools Help
http://boysen.bounceme.net
Google
ited Latest Headlines ES Seth JBMGR
eb
k
ile
OK
Cancel

John Beckett
Associate Professor of Computing
Southern Adventist University
P. O. Box 370, Collegedale, TN 37315

Authentication Required

A username and password are being requested by http://hw.cs.southern.edu. The site says: "HomeWork Server"

User Name: jbeckett
Password: ●●●●●●●●●●●

CPTE 109 Fall	Presentation Technology
CPTE 110 Fall	Intro to Web Development
CPTE 212 Fall	Web Programming

I'm working on a new

Lesson 13 – Application Builders

A growing trend on the Internet is application builder sites. You pay nothing in some cases, substantial amounts in others. This can be a viable option for a Website in some cases. But let's look at choice factors.

Advantages:

- Low bar to entry – cost may even be zero.
- Easy learning curve in many cases.
- High-powered tools give you advanced functionality.
- Integration with other "office" and e-commerce functions from the same vendor.

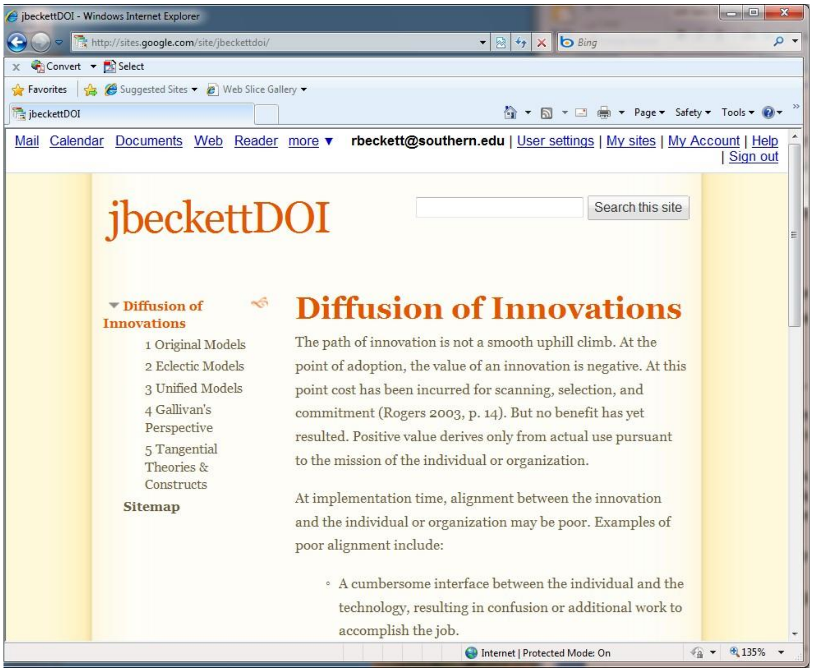
Disadvantages:

- Less-precise control of presentation.
- Tied in to whatever the vendor functionality (in this case Google) provides.
- If the vendor's system goes down, your site is down.

HW13: Implement the DOI site using Google Sites. You of course will be the creator. Invite your instructor to view the site. Things you'll have to do in order to accomplish this:

1. Obtain an account at Google.com.
2. Use sites.google.com to create your site.
   a. For a url, use your user name followed by "doi".
   b. Copy and paste from the DOI materials we're using.
   c. Make sure that the pages appear in order on the menu. I did this by numbering them. There will be extra credit if you can make them appear in order without using a sorting item as I did.
   d. Select a theme that results in a readable site.
3. Invite your jbeckett@southern.edu to view the site. Note: This is required for a grade!
4. Allow "anyone in the world" to view your site. Click "More Actions" in the upper right hand corner, then "Share this site."

Here's a picture of my page as viewed by rbeckett:


jbeckettDOI - Windows Internet Explorer
http://sites.google.com/site/jbeckettdoi/
Bing
Convert
Select
Favorites
Suggested Sites
Web Slice Gallery
jbeckettDOI
Page
Safety
Tools
Mail
Calendar
Documents
Web
Reader
more ▾
rbeckett@southern.edu | User settings | My sites | My Account | Help | Sign out
Search this site
Internet | Protected Mode: On
135%

jbeckettDOI

Diffusion of Innovations
1 Original Models
2 Eclectic Models
3 Unified Models
4 Gallivan's Perspective
5 Tangential Theories & Constructs
Sitemap

Diffusion of Innovations

The path of innovation is not a smooth uphill climb. At the point of adoption, the value of an innovation is negative. At this point cost has been incurred for scanning, selection, and commitment (Rogers 2003, p. 14). But no benefit has yet resulted. Positive value derives only from actual use pursuant to the mission of the individual or organization.

At implementation time, alignment between the innovation and the individual or organization may be poor. Examples of poor alignment include:

- A cumbersome interface between the individual and the technology, resulting in confusion or additional work to accomplish the job.

You may view this site at https://sites.google.com/site/jbeckettdoi/.

Lesson 14 – A Touch of JavaScript

This lesson will give you a taste – nay, a mere whiff – of something we hear about on the news these days: coding. A sample that is modest enough that you can survive even if this isn't your cup of tea. But enough so that a few of you will decide you want more. If so, welcome to the world of Computer Science!

The HTML and CSS we've written so far is what we call "static" – it doesn't do anything unless we click to load the next page. We could present just about any sort of information using this method. But whenever we click on new content, it requires that a new page be loaded.

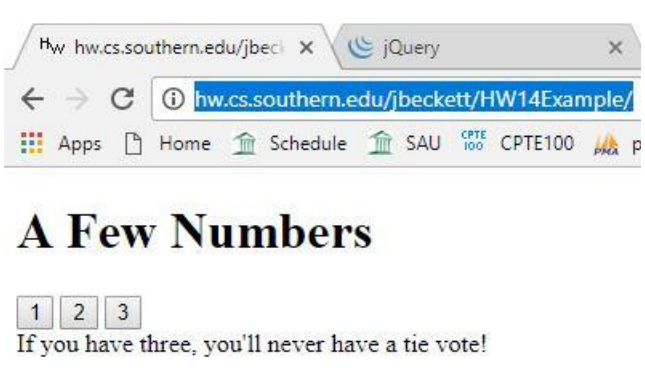
Adding some JavaScript code can make our pages much livelier. Without JavaScript, your browser simple accepts what the server sends it and shows it on the screen (with formatting for tags and CSS, of course). But with JavaScript running, the actual content of the page can change because of actions such as mouse clicks and position.

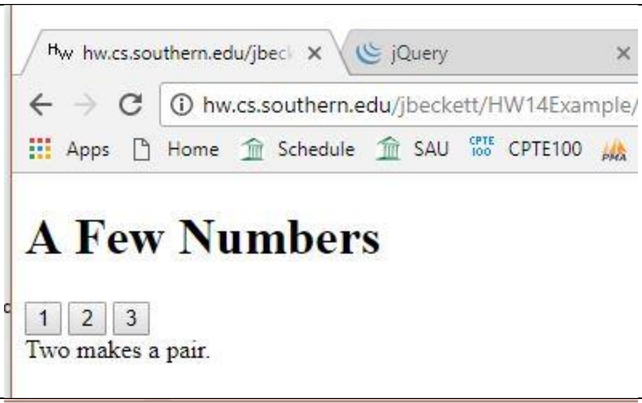
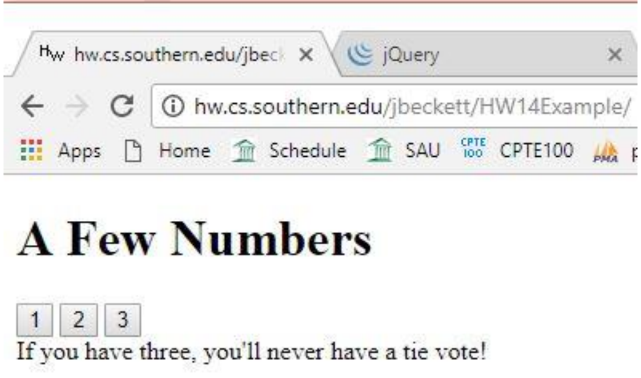
> ⚠ Just like HW09 when we were using server-side scripting, this assignment doesn't work when you click the html file on your computer. It must be loaded on a server.

There is a world of functionality we can add with JavaScript, but we'll be doing just one thing in this assignment: Replacing content on a page without re-loading the page, when the user clicks a button.

JavaScript code can be intense – but fortunately we don't have to write everything we might need. There are several *packages* people have written to do commonly-needed things in JavaScript, and they make these available to us. This assignment uses a package named "jquery" that is very popular in the industry. It is modest in size so it won't slow down page loading time much. The version we'll use is jquery-3.2.1.min.js. You can download it from jquery.com.

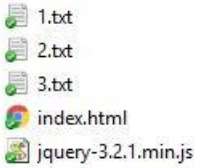
Our first example is an extremely simple page that illustrates the idea we'll implement in HW14. The page looks like this:

This is what the page looks like when initially loaded.	 hw.cs.southern.edu/jbec jQuery hw.cs.southern.edu/jbeckett/HW14Example/ Apps Home Schedule SAU CPTE100 1 2 3 A Few Numbers If you have three, you'll never have a tie vote!

If you click the “2” button, it looks like this.	 hw.cs.southern.edu/jbeck jQuery hw.cs.southern.edu/jbeckett/HW14Example/ Apps Home Schedule SAU CPTE100 A Few Numbers 1 2 3 Two makes a pair.
If you click the “3” button, it looks like this.	 hw.cs.southern.edu/jbec jQuery hw.cs.southern.edu/jbeckett/HW14Example/ Apps Home Schedule SAU CPTE100 A Few Numbers 1 2 3 If you have three, you'll never have a tie vote!

What is happening? When you click one of the numbers, the area below has its content replaced by the words in a file.

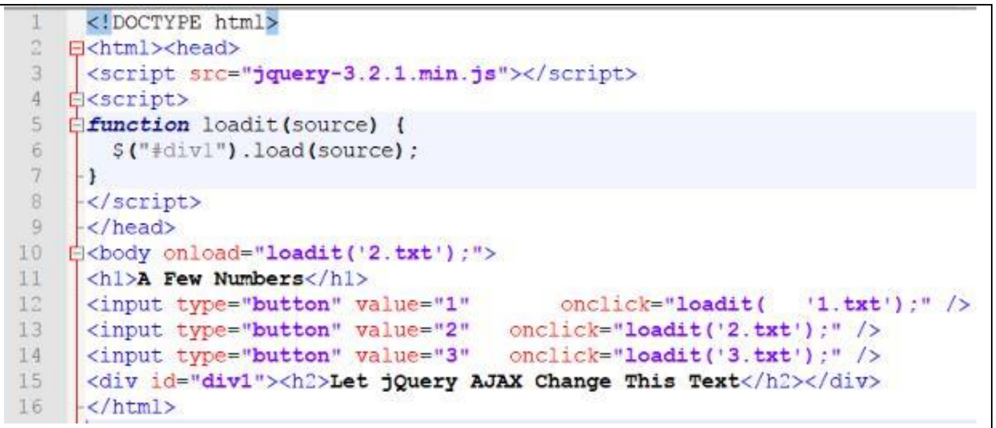
Now let’s look at the files required for this example.

- 1.txt is the text to plug in when the “1” button is clicked. - 2.txt is the text to plug in when the “2” button is clicked. - 3.txt is the text to plug in when the “3” button is clicked. - index.html is the page itself. - jquery-3.2.1.min.js is the JavaScript package that has functionality we are going to use.	 1.txt 2.txt 3.txt index.html jquery-3.2.1.min.js

Next, the code in index.html. New to us at this point are the <script> tag and the “on”.. attribute.

Interpretation:

3. Establish a link to the jquery package, which is located in the same directory
4. The <script> tag means we have JavaScript ahead. </script> means we’re done defining JavaScript.
5. Define a *function* that will do something if we *invoke* it later in the program.
6. This function will replace the HTML in the element (on line 15) with whatever is in the file mentioned in the *argument* named “source”.
10. A tag attribute beginning with “on” means we are about to have a very small JavaScript snippet that tells what to do when the event described after the letters “on” happens. In this case: When the page loads, we’ll use the “loadit()” function to load the text from the file named “2.txt”.
12. When they click the “1” button, ask loadit() to obtain and plug in the contents of “1.txt”. And so on for lines 13 and 14.
15. This is an HTML <div> element that initially contains some default text “Let jQuery AJAX Change This Text” – which is instantly replaced by that onload= attribute when the page loads.



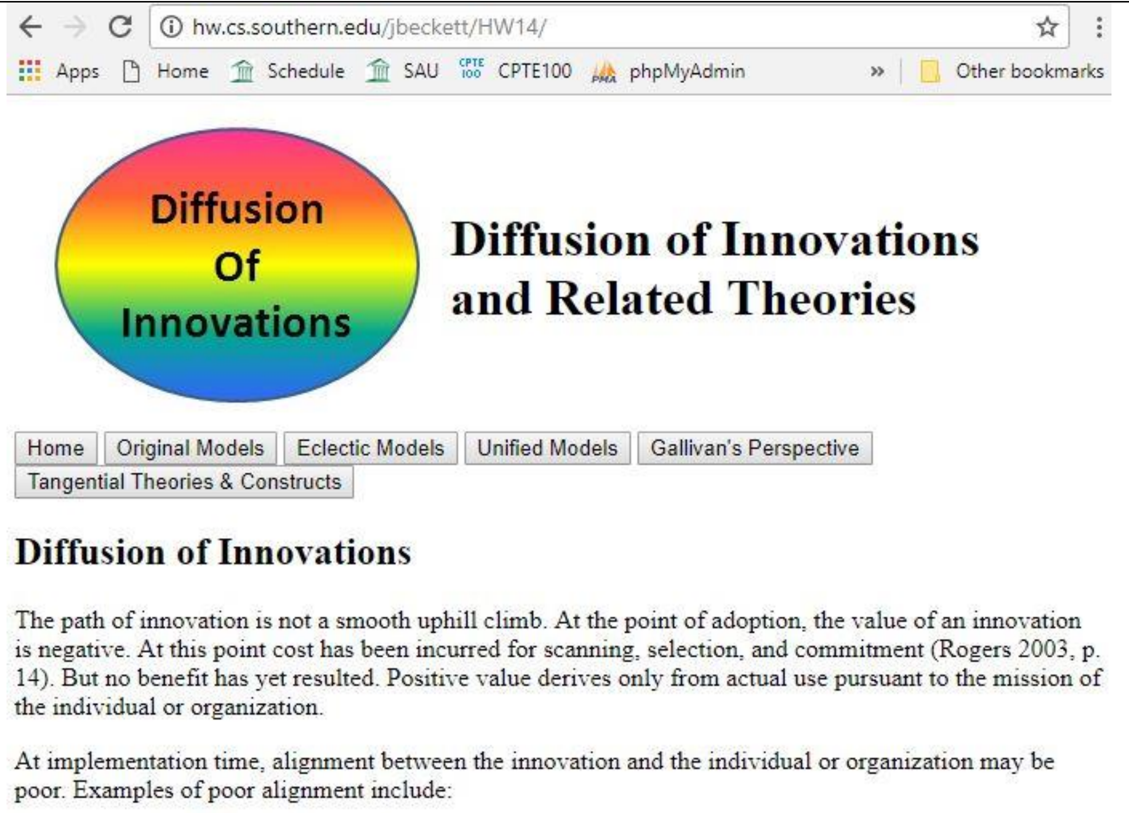
```
1    <!DOCTYPE html>
2   <html><head>
3   <script src="jquery-3.2.1.min.js"></script>
4   <script>
5   function loadit(source) {
6     $("#div1").load(source);
7   }
8   </script>
9   </head>
10  <body onload="loadit('2.txt');">
11  <h1>A Few Numbers</h1>
12  <input type="button" value="1"       onclick="loadit(   '1.txt');" />
13  <input type="button" value="2"   onclick="loadit('2.txt');" />
14  <input type="button" value="3"   onclick="loadit('3.txt');" />
15  <div id="div1"><h2>Let jQuery AJAX Change This Text</h2></div>
16  </html>
```

WARNING: Lines 6 and 15 both have a variable named “div1”. You can call it div-one or div-lower-case-ell and this example will work either way – but you must do it the same way both times!

One final point: We set the page up so that the initial contents of the display area had dummy text “Let jQuery…” in it. Why not include our desired text there as well as in the .txt file? Because then we would have had two copies of the information. It would have been too easy for someone needing to change that information, to change one and not the other. This is a variation of the philosophy: “Keep all your eggs in one basket, and take very good care of that basket.”

HW 14: Implement the DOI site on a single HTML page such that when the user clicks one of the menu buttons, the area for text will show the information requested.

This is the initial view of the page (only the top part shown.)



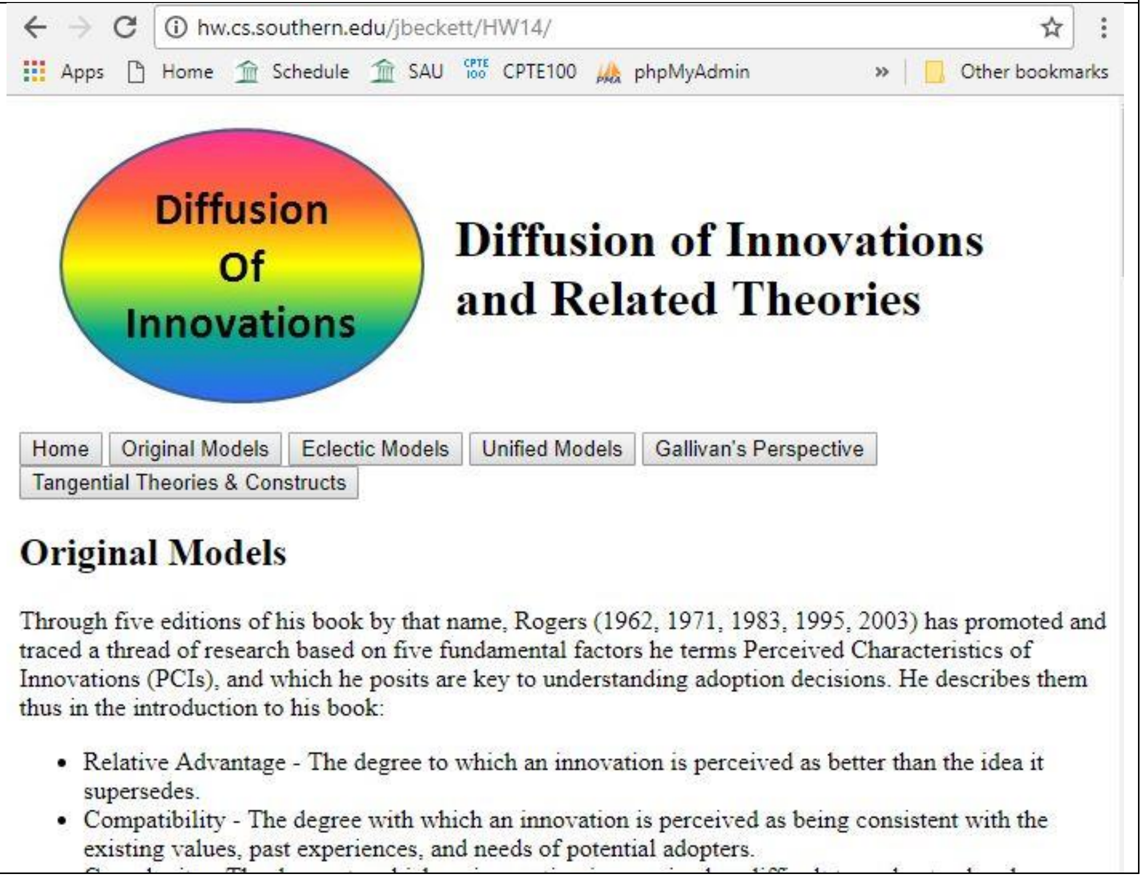
Diffusion of Innovations and Related Theories

Diffusion of Innovations

The path of innovation is not a smooth uphill climb. At the point of adoption, the value of an innovation is negative. At this point cost has been incurred for scanning, selection, and commitment (Rogers 2003, p. 14). But no benefit has yet resulted. Positive value derives only from actual use pursuant to the mission of the individual or organization.

At implementation time, alignment between the innovation and the individual or organization may be poor. Examples of poor alignment include:

When "Original Models" is clicked, it looks like this. Note that the URL is unchanged.

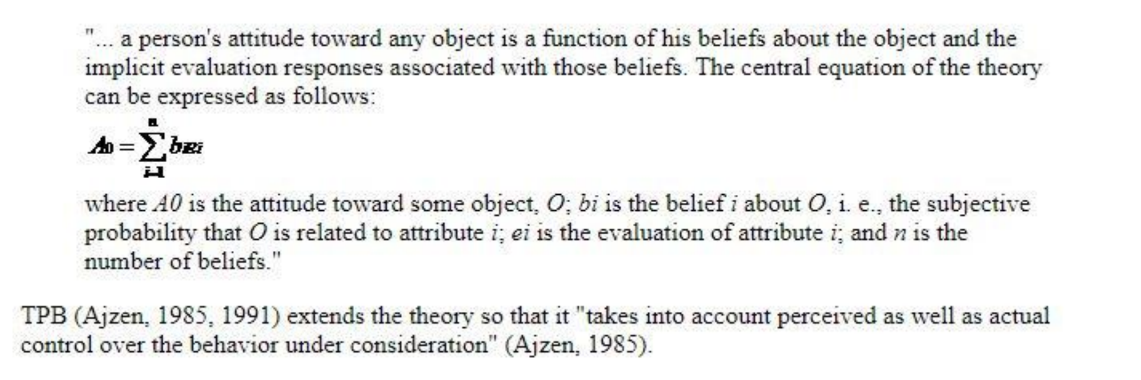
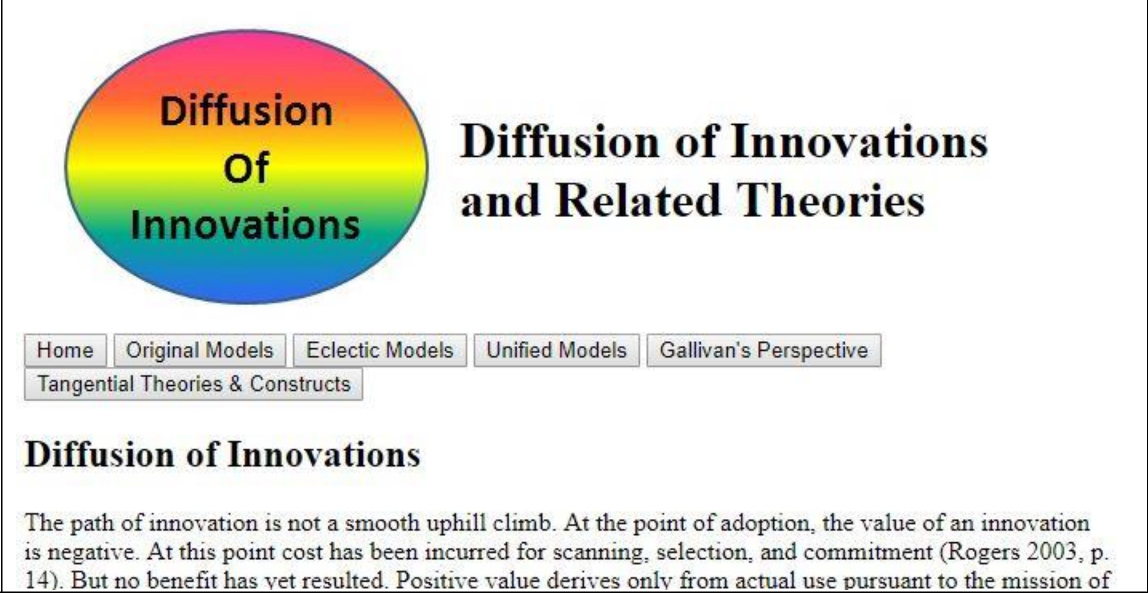


Diffusion of Innovations and Related Theories

Original Models

Through five editions of his book by that name, Rogers (1962, 1971, 1983, 1995, 2003) has promoted and traced a thread of research based on five fundamental factors he terms Perceived Characteristics of Innovations (PCIs), and which he posits are key to understanding adoption decisions. He describes them thus in the introduction to his book:

- Relative Advantage - The degree to which an innovation is perceived as better than the idea it supersedes.
- Compatibility - The degree with which an innovation is perceived as being consistent with the existing values, past experiences, and needs of potential adopters.

Scrolling down the page, we notice that the equation graphic show up just fine.	 "... a person's attitude toward any object is a function of his beliefs about the object and the implicit evaluation responses associated with those beliefs. The central equation of the theory can be expressed as follows: $A_0 = \sum_{i=1}^{n} b_i e_i$ where *A0* is the attitude toward some object, *O*; *bi* is the belief *i* about *O*, i. e., the subjective probability that *O* is related to attribute *i*; *ei* is the evaluation of attribute *i*; and *n* is the number of beliefs." TPB (Ajzen, 1985, 1991) extends the theory so that it "takes into account perceived as well as actual control over the behavior under consideration" (Ajzen, 1985).
Clicking the "Home" button takes us back to the beginning.	 Diffusion Of Innovations Home Original Models Eclectic Models Unified Models Gallivan's Perspective Tangential Theories & Constructs Diffusion of Innovations and Related Theories **Diffusion of Innovations** The path of innovation is not a smooth uphill climb. At the point of adoption, the value of an innovation is negative. At this point cost has been incurred for scanning, selection, and commitment (Rogers 2003, p. 14). But no benefit has yet resulted. Positive value derives only from actual use pursuant to the mission of

1. As always, start with a clean folder for HW14. Let's put our various materials in it.
2. Download jquery-3.2.1.min.js (the version number may change – go with it) from the eClass site or jquery.com and place it in the HW14 folder.
3. Use the copy-and-paste function to grab the text for the first page, then use Visual Studio Code to save that text in a file named "index.txt". You may not have to type the extension.
4. Do the same thing for files named "original.txt", "unified.txt" and so on.
5. Copy the images folder from the former site into HW14.
6. Type index.html from the example above. You'll be expanding it.
    - a. Add a title to the <head> section.
    - b. Keep the two script tags intact
        - i. The first connects with jquery.
        - ii. The second contains that function you'll need. Include all four lines beginning with <script> and ending with </script>
    - c. The <body> tag needs to be changed so that it loads "index.txt".
    - d. You'll be rewriting the buttons:
        - i. The value= attribute is the words you want to show on the button, e.g. "Home".
        - ii. The *parameter* you send loadit() will be the name of the text file to be loaded when that button is clicked. E.g. when they click the "Unified Models" button the argument should be "unified.txt".
7. One you have it complete, upload the assignment to the server and run it from there.

⚠ You should know everything you need to know at this point to do this assignment. But perhaps you don't see a path to success. This is a common problem when one is learning to code. It's a lot like when you learned to drive: There is a bunch of thinking (psychologists call it "sequencing") involved in connecting the various things you do when driving with the overall process of making an automobile do what it does, and connecting all that with information from a map. Not to worry: a sequence for accomplishing this assignment is right here!

Conclusion

In these lessons we've learned how to code some basic HTML and CSS. Hopefully, by this time you can look up new functions if you need something a bit outside of what we've done.

We've also covered forms – something you'll need if you are collecting information. What to do with that information is for another class, CPTE 212.

We have used WordPress, a popular Content Management System. In the process we've seen some of what this technology can do for us, and how HTML and CSS skills help us get even more out of our system.

A brief trip into DNS technology has showed us how to establish a connection between our site and visitors out there on the Web. Follow the sequence in lesson 12 and you should find it unnecessary to register with a search engine – to say nothing of actually pay money for listings.

Application Builders are actually Content Management Systems – except that they are very friendly, often free, and perhaps a little less flexible. There will be times when we want to use them.

Finally, we saw how JavaScript can make a site much livelier (and by extension, operate with less load on the server). And JavaScript doesn't require us to be programmers! We can often use programs already written just by adjusting come configuration files.

I hope you've accomplished your goal of getting started on Web Development, and that some of you will decide that programming is so exciting you just have to get more. See you in the next class!

Appendix A – Useful Links

This book's Web page, with resources you'll need http://computing.southern.edu/jbeckett/WebDev1/

Official source of Web information http://www.w3.org/

Information on media embedding http://www.mediacollege.com/video/streaming/embed/

Index of HTML tags from W3Schools http://www.w3schools.com/tags/default.asp

Introduction to CSS from HTML.net http://www.html.net/tutorials/css/

Introduction to CSS from W3Schools http://www.w3schools.com/css/css_intro.asp

Many examples of CSS rules from W3Schools http://www.w3schools.com/css/css_examples.asp