CIS 4004: Web Based Information Technology
Fall 2012

An Introduction To HTML5

Instructor : Dr. Mark Llewellyn
markl@cs.ucf.edu
HEC 236, 407-823-2790
http://www.cs.ucf.edu/courses/cis4004/fall2012

Department of Electrical Engineering and Computer Science
University of Central Florida

Overview of Markup Languages

- A markup language is simply a set of rules that defines the layout, format, or structure of text within a document.
- After markup instructions are added to a document, the document must be read, or processed, by a program that knows how to interpret the markup elements.
- Markup languages existed long before the World Wide Web. They were used primarily in the publishing industry prior to their adaptation to computer programming.
- Work began in the 1960s to develop a standardized document markup language that would be platform independent.

Overview of Markup Languages

- The Standard Generalized Mark Language (SGML) was the result of this initiative and was the first standardized markup language to gain acceptance.

- It wasn’t until the Web exploded in popularity in the mid-1990s that the benefits of an open standard for markup languages became overwhelmingly apparent.

Overview of Markup Languages

- SGML is the ancestor of, and provides the framework for, current Web markup languages, including XHTML, XML, and HTML.

- SGML was developed as a markup language for large documents, such as technical documentation.

- It was adopted as an international standard by the International Organization for Standards (ISO) in 1986, and has been widely used by many industries, including the automotive industry, the health care industry, the IRS, and the US Department of Defense, for large scale documentation projects.

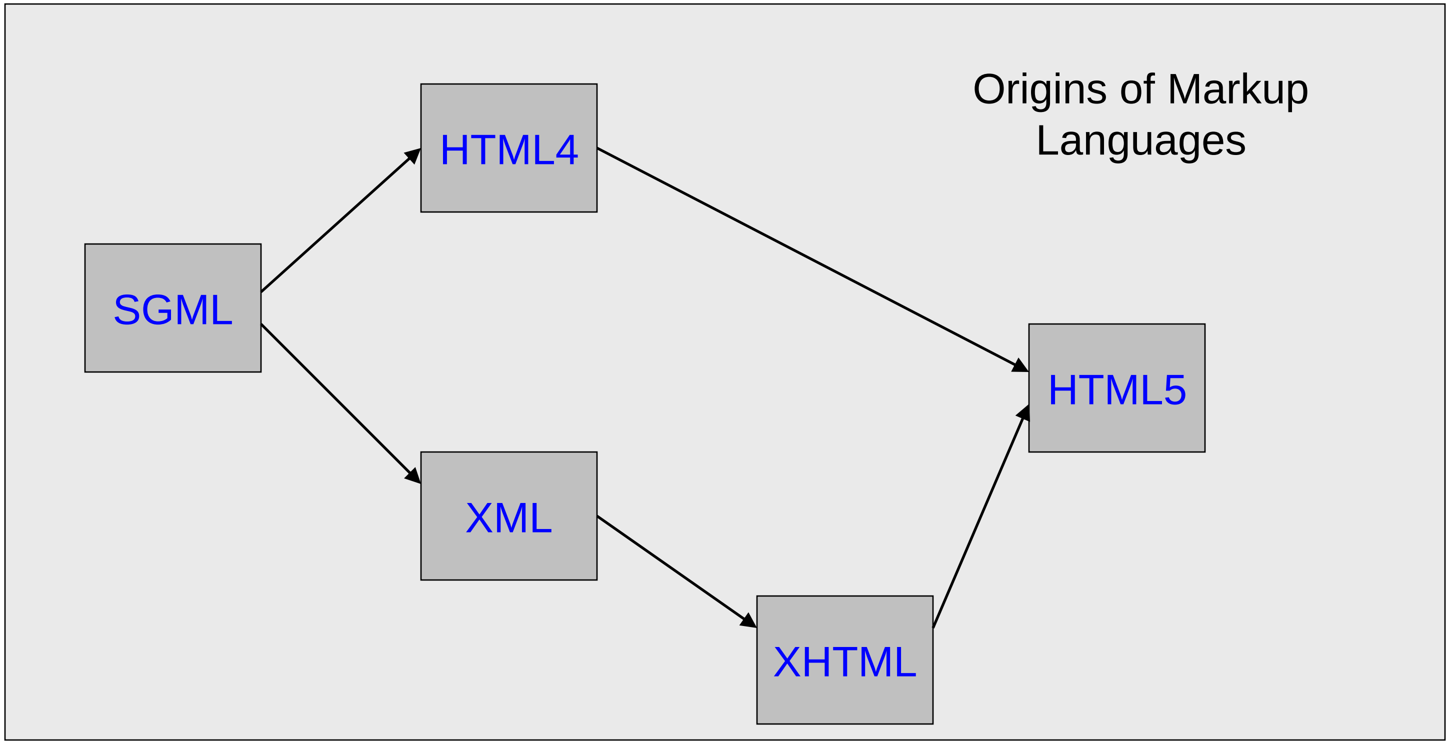
- SGML is extremely complex, and thus very expensive. SGML has proved useful mainly to organizations that have the expertise and budget to implement the expansive SGML specification.

Overview of Markup Languages

- When the World Wide Web was in its infancy in the late 1980s and early 1990s, SGML was the perfect tool for building the markup language that would be used to create documents for this new medium.
- Hypertext Markup Language (HTML) was developed as a lightweight SGML by researchers at CERN (European Organization for Nuclear Research) in the early 1990s.
- CERN had been involved with working on the SGML specification for many years.
- HTML was much smaller than SGML and gained widespread acceptance very quickly. It provided content developers with a portable document format that was not tied to any particular program or platform, and being an open standard, it was completely free to use.

Overview of Markup Languages

- HTML was adopted shortly after its development by the W3C (www.w3c.org), which continues to maintain the HTML specification, currently at version 5.



Overview of Markup Languages

- Because HTML documents are simple text documents embedded with markup elements, they are completely portable across platforms and programs.
- HTML documents can be displayed using any program running on any operating system that knows how to interpret the HTML language.
- This gives developers an incredible amount of flexibility and allows them to move files freely among platforms and programs.
  - For example, and HTML file created with a Mac text editor would look the same when open in a Windows text editor, and would be displayed the same when viewed using Netscape Navigator or Internet Explorer, either on a Mac or a PC.

Limitations of HTML

- As web technologies continue to advance at a very rapid pace, HTML has been pushed to its limits by developers and vendors.
- Somewhat ironically, the traits of HTML that helped build its popularity – its small size, limited number of elements, and ease of use – have become its downfall.
- HTML is a fixed specification with a finite set of elements. It is not extendable, and as a result of this limitation, Web developers and software vendors have stretched the usefulness of HTML almost to a breaking point.
- Browser vendors such as Microsoft and Netscape, have added proprietary features and additional HTML elements to their browsers based on demands for more functionality, but in so doing, they have compromised one of the most important benefits that HTML has to offer – portability. Given these proprietary additions to particular browsers, HTML pages developed for use in one Web browser may not display the same way when displayed in another browser or on another platform.

Limitations of HTML

- As a result of these limitations, it was expected that HTML 4.01 would be the last version of HTML as XHTML supplanted it in 2000.
- The first version, and only version of XHTML, 1.0, was released by the W3C in 2000 as the successor to HTML. This is the reason that the terms HTML and XHTML are often used interchangeably.
- Extensible Hypertext Markup Language (XHTML) is the modern “version” of HTML.
- XHTML 1.0 was the standard adopted by W3C in 2002.

XML – The Future of Web Markup Languages

- The limitations of HTML made it very clear that a new and better language was necessary for formulating Web documents.

- In addition, many companies were adding transactional functionality to their Web sites, such as allowing visitors to purchase items and services online.

- This marked a radical departure from the first generation of websites, which mainly provided static information that was easily stored as text. These new websites relied heavily on data gathered from different sources, such as databases, news feeds, and other Web sites.

- The resulting language was the Extensible Markup Language (XML). XML is an extensible language because, unlike HTML, it allows users to define their own tags.

XML – The Future of Web Markup Languages

- The first recommendation, XML 1.0, was released in 1998.
- The XML family of technologies was developed to separate document data from presentation and to give developers the ability to extend the element sets of XML languages as needed.
- XML itself is **not** a language – it is a meta language. A meta language is a set of rules used for building markup languages.
- Structured languages can be developed that describe certain types of data rather than just the presentation of the data. Such structured languages include elements that describe documents containing information about an account, an item, a service, or a transaction.
- XHTML is an application of XML that is used for formatting Web documents. There are many other XML languages, some still under development, such as RSS (Really Simple Syndication), MathML, GraphML, Scalable Vector Graphics, and MusicXML.

Some Of The Benefits Of XML

- It allows data to be self-describing, as opposed to being limited by a pre-defined set of elements.
- You can provide rules for XML elements that limit the type of data an element can contain, such as only letters, only numbers, or a certain number of characters.
- Lets you create custom data structures for industry-specific or company-specific needs.
- Because XML describes data, you can present the data any number of ways by applying different presentation styles.
- It provides a rich set of tools for linking.
- You can use it to interchange data between proprietary formats and between databases or data structures.
- You can use the tools to defines a standard syntax for many markup languages.
- It has much more robust and reliable data searching capabilities that HTML.

Some Of The Benefits Of XML

- As its name implies, XHTML is extensible, meaning that its element set is not finite like the element set of HTML.

- Like any XML language, XHTML can be extended to add elements if needed or it can incorporate elements from various XML languages into its element set.

- Keep in mind, however, that if your goal is to keep XHTML compatible with current browsers (certainly one of our goals in this class), you may not be able to use some of its more advanced features. But if your documents are written in XHTML now, you will be able to integrate these features with ease as soon as they become mainstream.

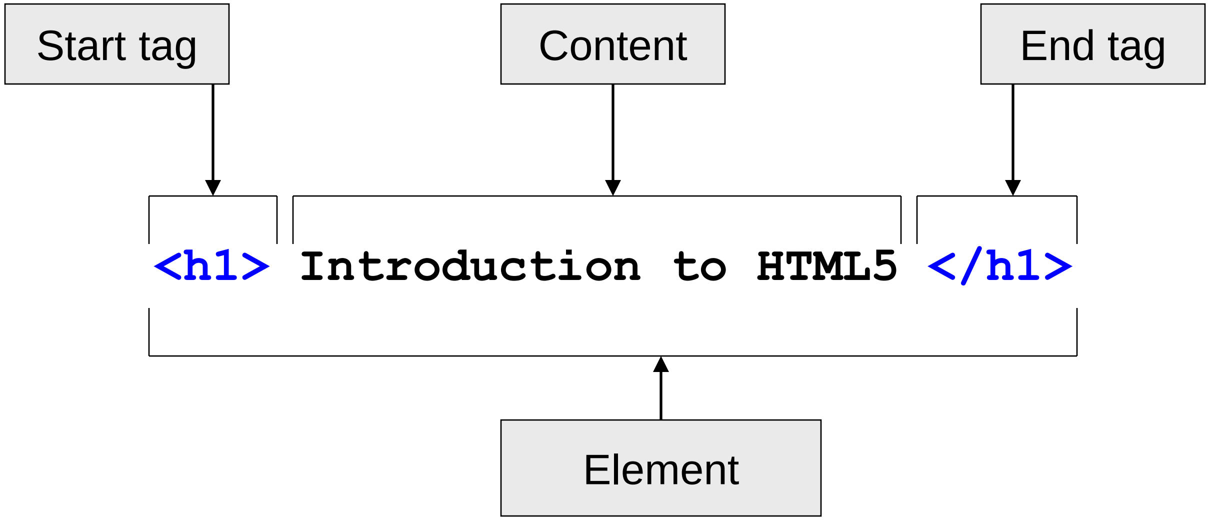
XHTML Document Building Blocks

- Like any language, XHTML has a number of building blocks that are used to create complete documents.
  - The English language, for example, is made up of nouns, verbs, adjectives, adverbs, prepositions, and so on, which are used in conjunction with each other to form sentences and paragraphs.
- Both HTML and XHTML provide language building blocks that can be added to any text document. Web browsers know how to interpret these elements in order to present the document based on formatting rules.

XHTML Elements

- XHTML elements are the core components of XHTML documents and are used to describe the data in a document.
  - Elements are like nouns in the English language.
- Elements are the markup, or formatting instructions, of the XHTML document.
- Elements define the text styles, formatting links, and other pieces of the document.
- Elements and tags are sometimes used interchangeably, but strictly speaking, a tag is a piece of an element.

HTML5 Elements



- All elements, except for empty elements, consist of three pieces: a start tag, content, and an end tag.
- HTML5 element names by convention are written in lowercase letters.

Empty Elements

- Empty elements are used primarily to describe pieces of data that don’t contain any content.
- For example, some common empty elements in HTML5 are:

  `<br>` for line break

  `<img>` for image

  `<p>` for paragraph

- In XML and HTML5, all elements must have a start tag and end tag.
- The XML specification provides a shortcut for writing an empty element using a single tag. This is shown below:

  `<br />` for line break

  `<img />` for image

  `<p />` for paragraph

NOTE: The practice of adding the extra space is **not** part of the syntax of either XML or HTML5. The reason for the space it to make this code compatible with older versions of Web browsers. Because HTML does not require end tags for these elements, most older browsers do not know how to handle the new empty element syntax. The extra space allows these browsers to interpret the syntax correctly.

Attributes

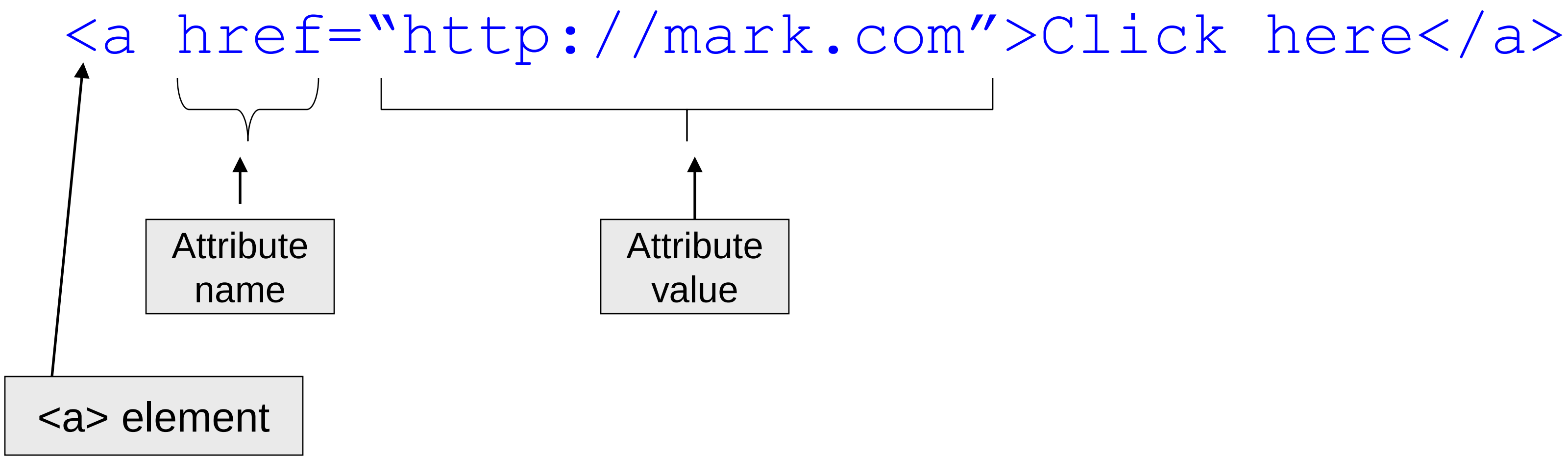
- HTML5 attributes are pieces of information that help to describe elements. Some elements have required attributes, others are optional and depend on the content that is being marked up.
- Attributes are referred to as name-value pairs and have the following syntax: The name of the attribute is on the left, followed by an equal sign, then the value.

```
    name = “value”
```

- Here are a few examples:

```
<a href=“http://mark.com”>Click here</a>

<img src=“/images/pictures.gif” id=“Picture of House”  />
```

Attributes

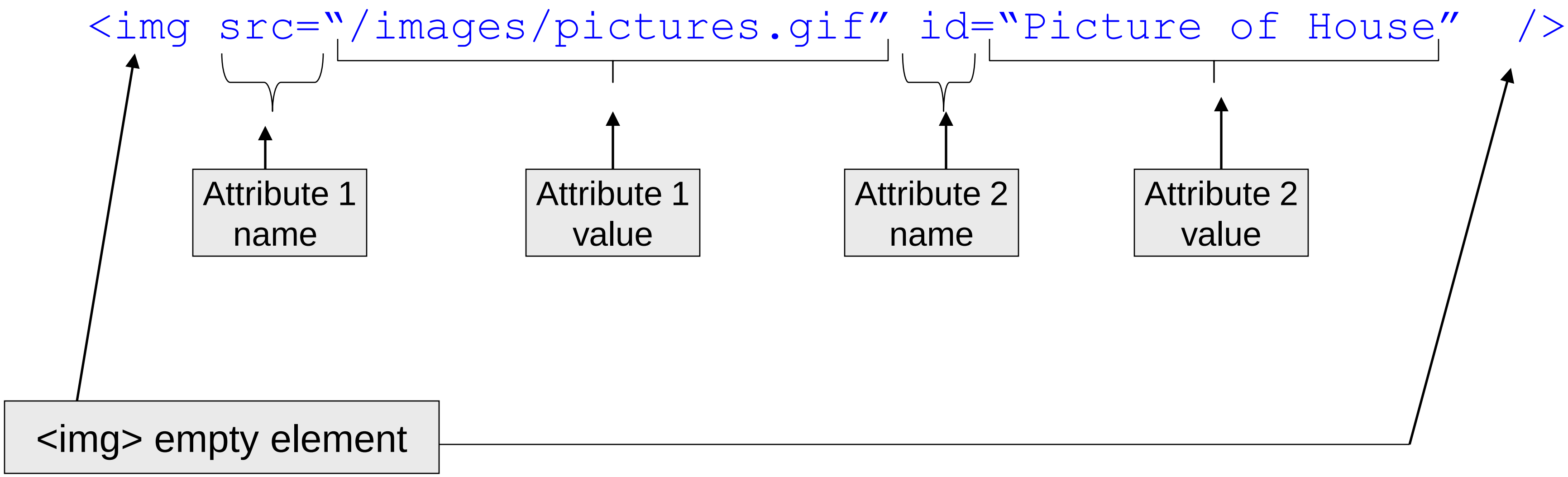


```
<a href=“http://mark.com”>Click here</a>
```

Attribute name

Attribute value

<a> element



```
<img src=“/images/pictures.gif” id=“Picture of House”  />
```

Attribute 1 name

Attribute 1 value

Attribute 2 name

Attribute 2 value

<img> empty element

Attribute Rules

- We’ll see more rules later, but here are a few rules about HTML5 attributes:

1. Attributes are always contained within the start tag of an element.

2. Names must be in lowercase letters.

3. Attributes must have a value that is surrounded by quotes.

HTML Core Attributes

- In the HTML5 specification, there are a set of core attributes that can be used with most XHTML elements.
- They cannot be used in `<base>`, `<head>`, `<html>`, `<meta>`, `<param>`, `<script>`, `<style>`, and `<title>` elements.
- The core attributes are:

1. **`id`** – document wide unique id.
2. **`class`** – list of classes of the element.
3. **`style`** – associated style information.
4. **`title`** – advisory title amplification.

HTML5 Comments

- Comments in HTML5 are notations that are ignored by programs and parsers.

- The syntax of an HTML5 comment is identical to XHTML and XML comments.

- You can use comments to document your code, add additional information about a piece of data, add visual breaks, or add information that other people working on or using your document would find useful.

- The following is an HTML5 comment:

```
<!--  This is a comment  -->
```

Creating Your First HTML5 Document

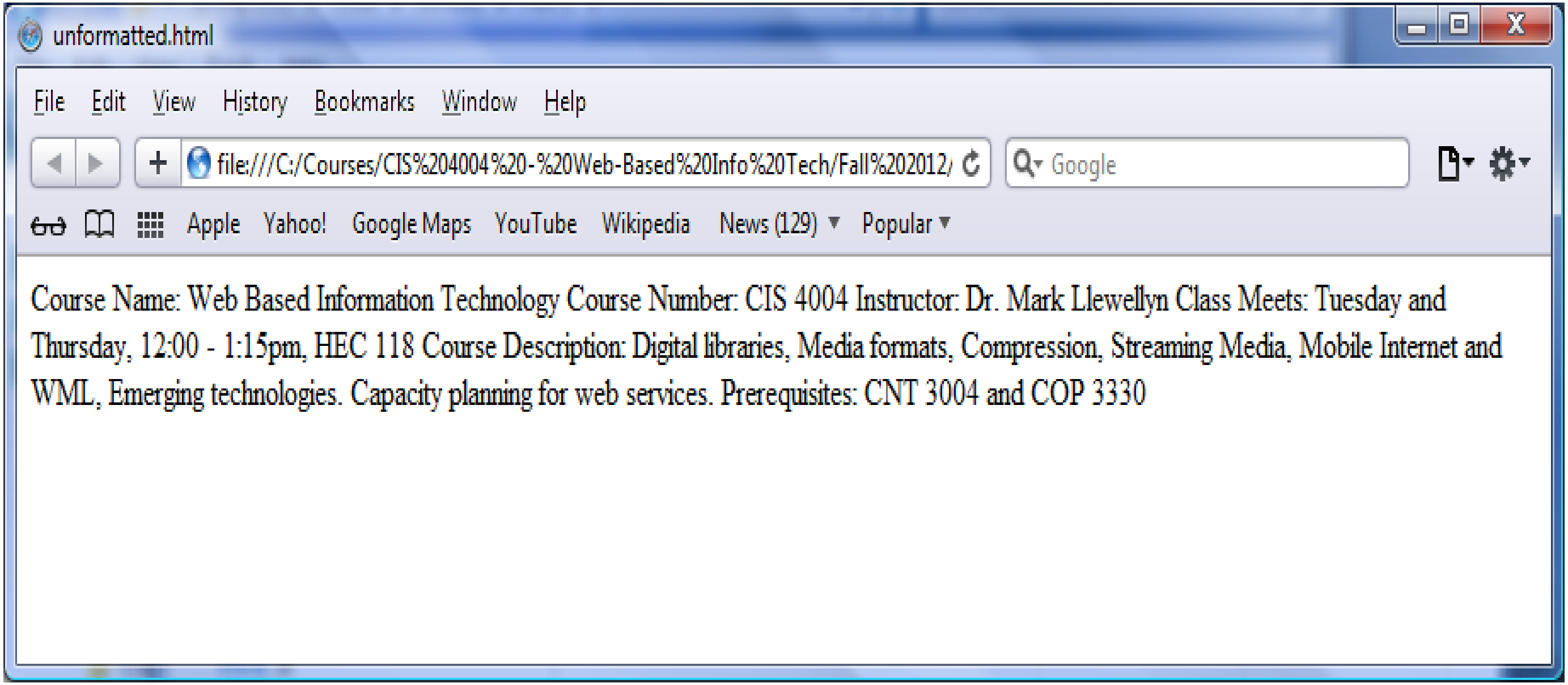
- We’ll begin creating our first HTML5 document by examining a plain text document that we would like to markup with formatting elements.
- Using NotePad or some similar text editor, create a file with the name “`unformatted.html`”, that includes the text shown below (type it exactly as it appears below):

```
Course Name:  Web Based Information Technology
Course Number: CIS 4004
Instructor:  Dr. Mark Llewellyn
Class Meets:  Tuesday and Thursday, 12:00 – 1:15 pm, HEC 118

Course Description: Digital libraries, Media formats, Compression, Streaming Media,
            Mobile Internet and WML, Emerging technologies.  Capacity planning for web
services.

Prerequisites: CNT 4004 and COP 3330
```

Unformatted Text Document – name it “unformatted.html”

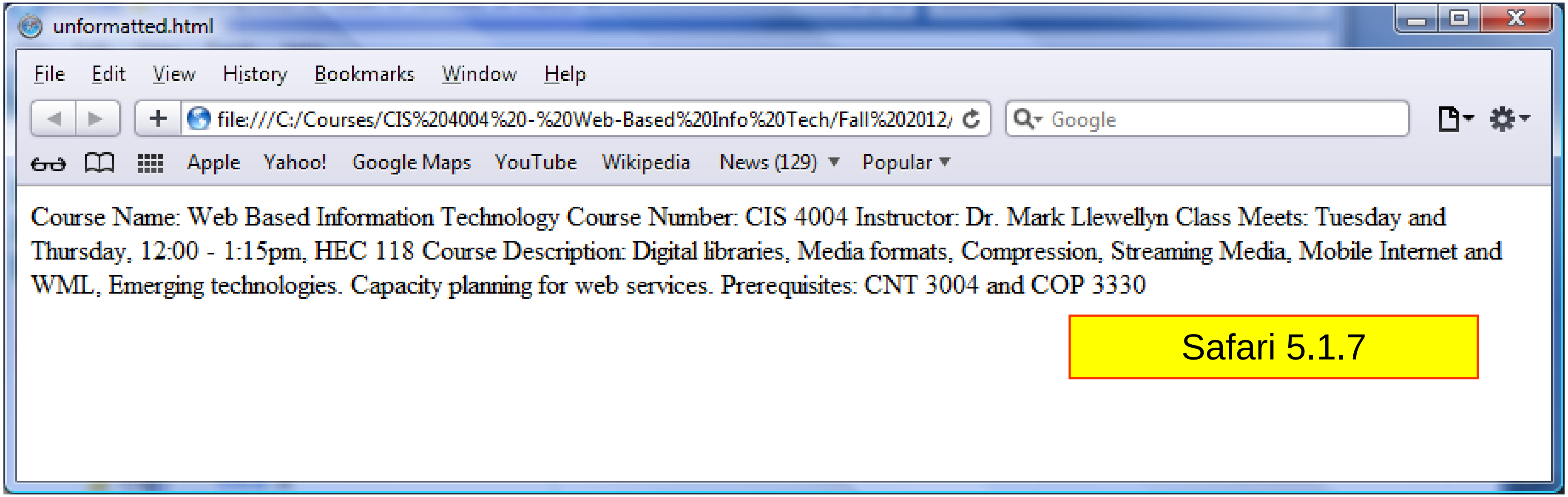
Creating Your First XHTML Document

- Next, open your Web browser and load the file “unformatted.html” that you just created. The screen shot below, shows what the file looks like in Internet Explorer 8. The following page show the same file in two other popular browsers.



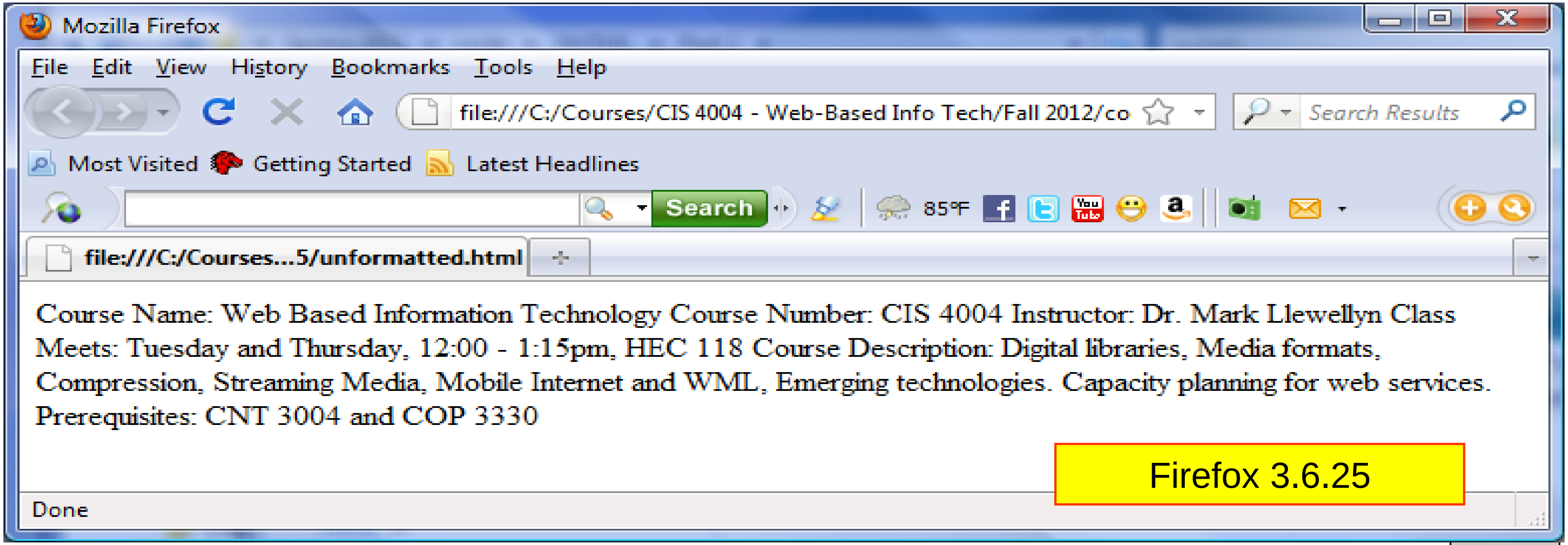
unformatted.html
File Edit View History Bookmarks Window Help
file:///C:/Courses/CIS%204004%20-%20Web-Based%20Info%20Tech/Fall%202012/
Google
Apple Yahoo! Google Maps YouTube Wikipedia News (129) Popular

Course Name: Web Based Information Technology Course Number: CIS 4004 Instructor: Dr. Mark Llewellyn Class Meets: Tuesday and Thursday, 12:00 - 1:15pm, HEC 118 Course Description: Digital libraries, Media formats, Compression, Streaming Media, Mobile Internet and WML, Emerging technologies. Capacity planning for web services. Prerequisites: CNT 3004 and COP 3330


unformatted.html
File Edit View History Bookmarks Window Help
file:///C:/Courses/CIS%204004%20-%20Web-Based%20Info%20Tech/Fall%202012/
Google
Apple Yahoo! Google Maps YouTube Wikipedia News (129) ▾ Popular ▾

Course Name: Web Based Information Technology Course Number: CIS 4004 Instructor: Dr. Mark Llewellyn Class Meets: Tuesday and Thursday, 12:00 - 1:15pm, HEC 118 Course Description: Digital libraries, Media formats, Compression, Streaming Media, Mobile Internet and WML, Emerging technologies. Capacity planning for web services. Prerequisites: CNT 3004 and COP 3330

Safari 5.1.7


Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///C:/Courses/CIS 4004 - Web-Based Info Tech/Fall 2012/co
Search Results
Most Visited Getting Started Latest Headlines
Search
85°F
file:///C:/Courses...5/unformatted.html
Done

Course Name: Web Based Information Technology Course Number: CIS 4004 Instructor: Dr. Mark Llewellyn Class Meets: Tuesday and Thursday, 12:00 - 1:15pm, HEC 118 Course Description: Digital libraries, Media formats, Compression, Streaming Media, Mobile Internet and WML, Emerging technologies. Capacity planning for web services. Prerequisites: CNT 3004 and COP 3330

Firefox 3.6.25

Creating Your First HTML5 Document

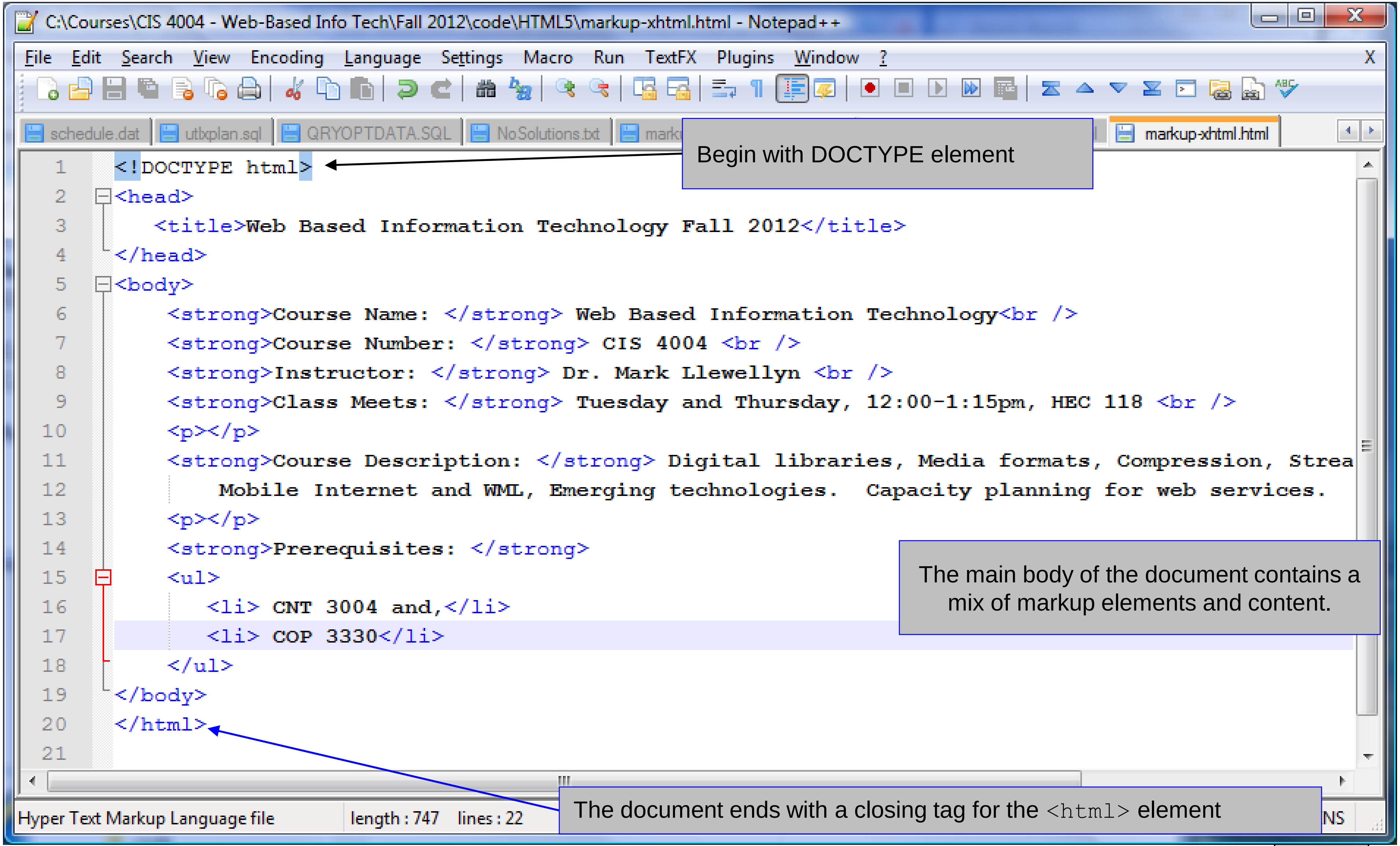
- The screen shot on the previous page illustrates how the Web browser can open the document, but without formatting instructions, it does not know how to correctly format the document.

- Web browsers ignore all whitespace characters, including line breaks, so without the proper markup, our document displays as just a block of text.

- Next, we'll add some formatting elements (markup) to our document.

- **NOTE:** Do not type the line numbers in your document, they are for only used in these notes to make references to specific lines.

The HTML5 Markup



C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\markup-xhtml.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
schedule.dat
utlxplan.sql
QRYOPTDATA.SQL
NoSolutions.txt
mark
markup-xhtml.html
Hyper Text Markup Language file
length : 747 lines : 22
NS

```
<!DOCTYPE html>
<head>
   <title>Web Based Information Technology Fall 2012</title>
</head>
<body>
    <strong>Course Name: </strong> Web Based Information Technology<br />
    <strong>Course Number: </strong> CIS 4004 <br />
    <strong>Instructor: </strong> Dr. Mark Llewellyn <br />
    <strong>Class Meets: </strong> Tuesday and Thursday, 12:00-1:15pm, HEC 118 <br />
    <p></p>
    <strong>Course Description: </strong> Digital libraries, Media formats, Compression, Strea
        Mobile Internet and WML, Emerging technologies.  Capacity planning for web services.
    <p></p>
    <strong>Prerequisites: </strong>
    <ul>
       <li> CNT 3004 and,</li>
       <li> COP 3330</li>
    </ul>
</body>
</html>
```

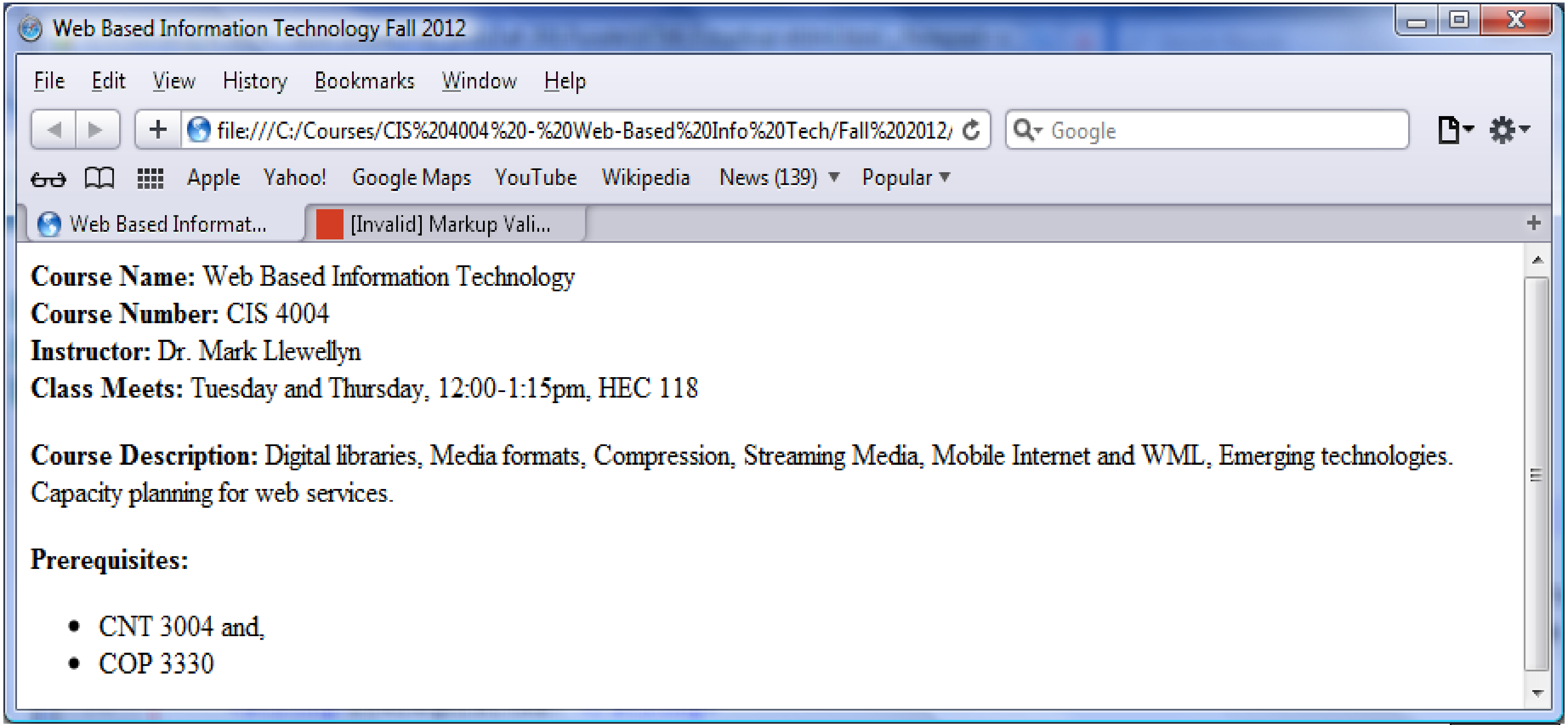
Begin with DOCTYPE element

The main body of the document contains a mix of markup elements and content.

The document ends with a closing tag for the `<html>` element

Rendering/Viewing The HTML5 Markup

- Create a new file that contains the text from the previous page (remember – do not type in the line numbers). I called my version of this file: "`markup-xhtml.html`", but you can call it whatever you would like. Next, open your Web browser and load the file "`markup-xhtml.html`" that you just created. The screen shot below, shows what the file looks like in Safari.



Web Based Information Technology Fall 2012
File Edit View History Bookmarks Window Help
file:///C:/Courses/CIS%204004%20-%20Web-Based%20Info%20Tech/Fall%202012/
Google
Apple Yahoo! Google Maps YouTube Wikipedia News (139) Popular
Web Based Informat...
[Invalid] Markup Vali...

Course Name: Web Based Information Technology
Course Number: CIS 4004
Instructor: Dr. Mark Llewellyn
Class Meets: Tuesday and Thursday, 12:00-1:15pm, HEC 118

Course Description: Digital libraries, Media formats, Compression, Streaming Media, Mobile Internet and WML, Emerging technologies. Capacity planning for web services.

Prerequisites:

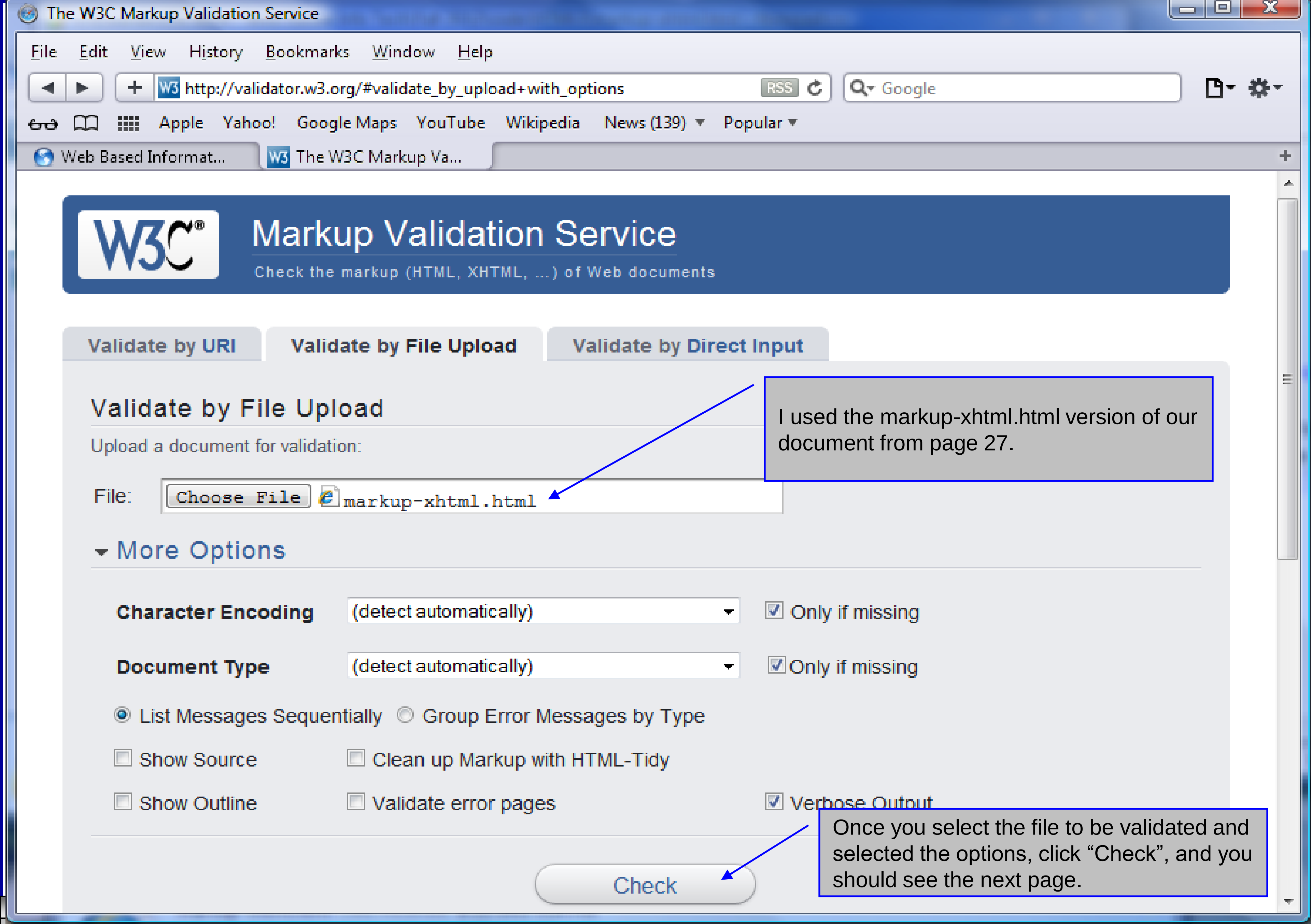
- CNT 3004 and,
- COP 3330

Validating An HTML5 Document

- While there are many free and commercial programs available, one of the simplest ways to check if your XHTML documents are well-formed and valid is to use the free on-line validating tool from the W3C Web site. It can check your document either from your computer or from a website.

- The W3C markup validation service is available at: http://validator.w3.org.

- The next page shows the first step in using the file upload version of the validator.


The W3C Markup Validation Service
File Edit View History Bookmarks Window Help
http://validator.w3.org/#validate_by_upload+with_options
RSS
Google
Apple Yahoo! Google Maps YouTube Wikipedia News (139) Popular
Web Based Informat...
The W3C Markup Va...
W3C®

Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Validate by URI
Validate by File Upload
Validate by Direct Input

Validate by File Upload

Upload a document for validation:

File: Choose File markup-xhtml.html

More Options

Character Encoding (detect automatically) ☑ Only if missing

Document Type (detect automatically) ☑ Only if missing

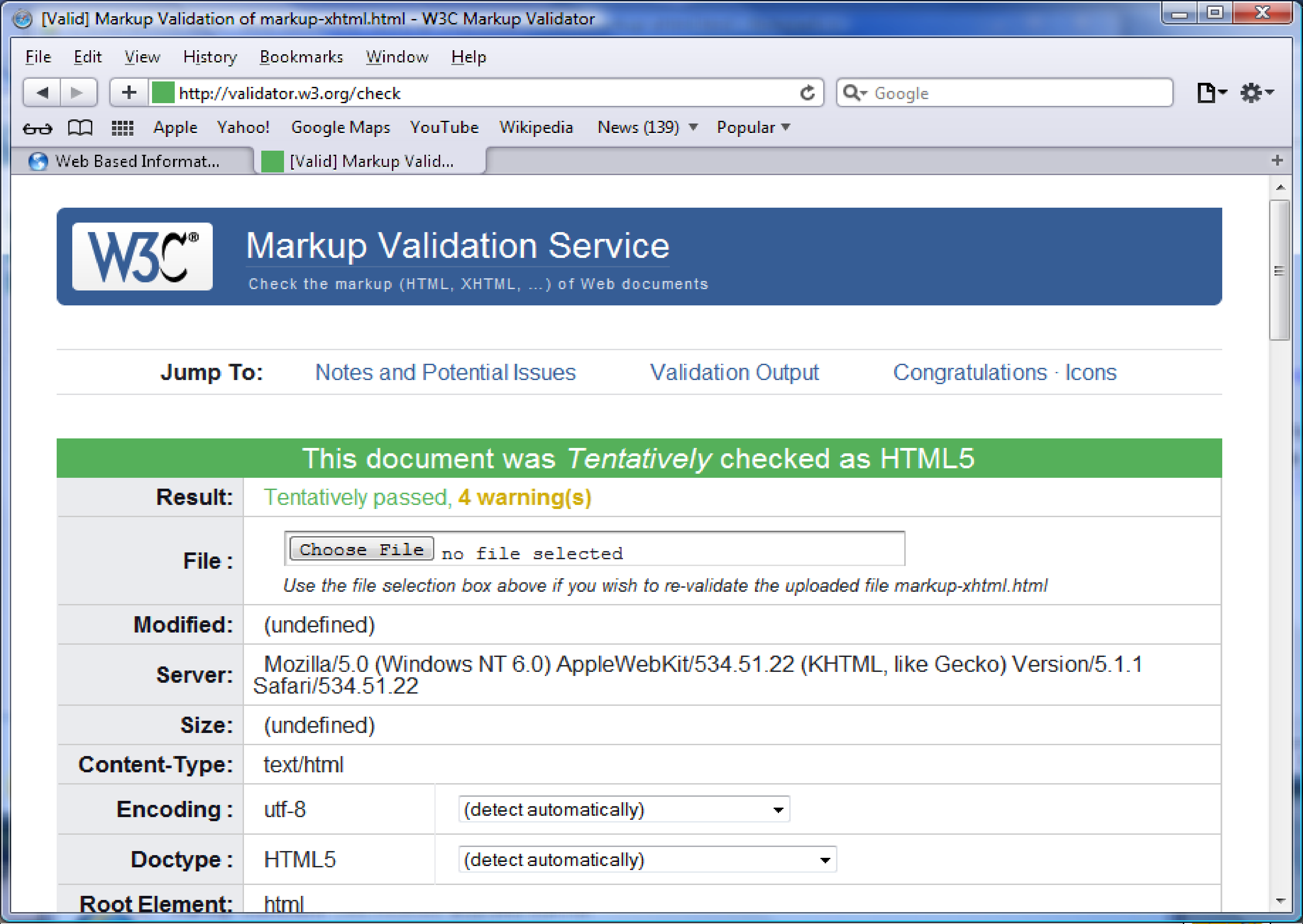
● List Messages Sequentially ○ Group Error Messages by Type

□ Show Source □ Clean up Markup with HTML-Tidy

□ Show Outline □ Validate error pages ☑ Verbose Output

I used the markup-xhtml.html version of our document from page 27.

Once you select the file to be validated and selected the options, click “Check”, and you should see the next page.



[Valid] Markup Validation of markup-xhtml.html - W3C Markup Validator
File Edit View History Bookmarks Window Help
http://validator.w3.org/check
Google
Apple Yahoo! Google Maps YouTube Wikipedia News (139) Popular
Web Based Informat...
[Valid] Markup Valid...

Markup Validation Service

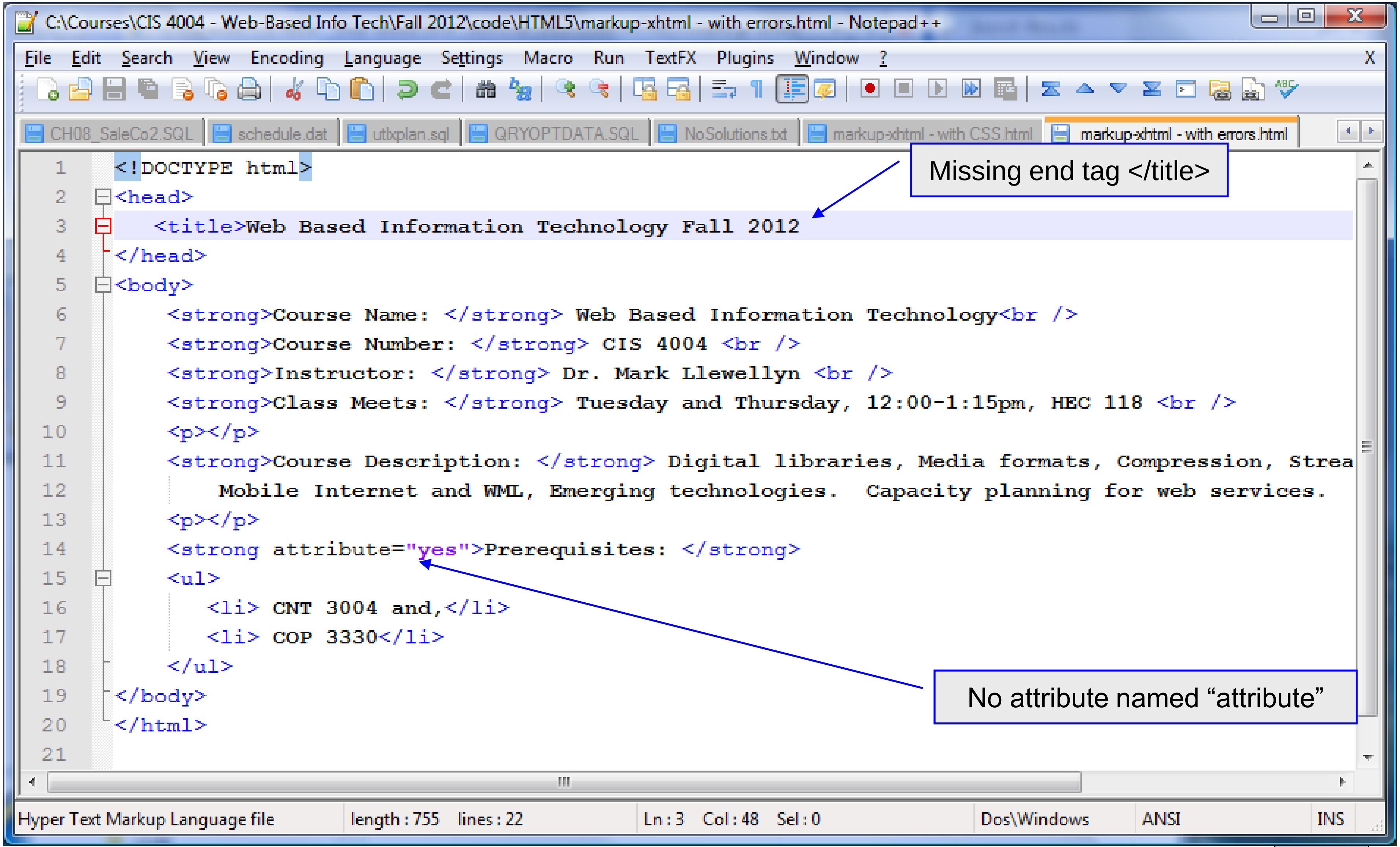
Check the markup (HTML, XHTML, ...) of Web documents

Jump To: Notes and Potential Issues | Validation Output | Congratulations · Icons

This document was *Tentatively* checked as HTML5

Result:	Tentatively passed, **4 warning(s)**	
File :	Choose File no file selected *Use the file selection box above if you wish to re-validate the uploaded file markup-xhtml.html*	
Modified:	(undefined)	
Server:	Mozilla/5.0 (Windows NT 6.0) AppleWebKit/534.51.22 (KHTML, like Gecko) Version/5.1.1 Safari/534.51.22	
Size:	(undefined)	
Content-Type:	text/html	
Encoding :	utf-8	(detect automatically)
Doctype :	HTML5	(detect automatically)
Root Element:	html	

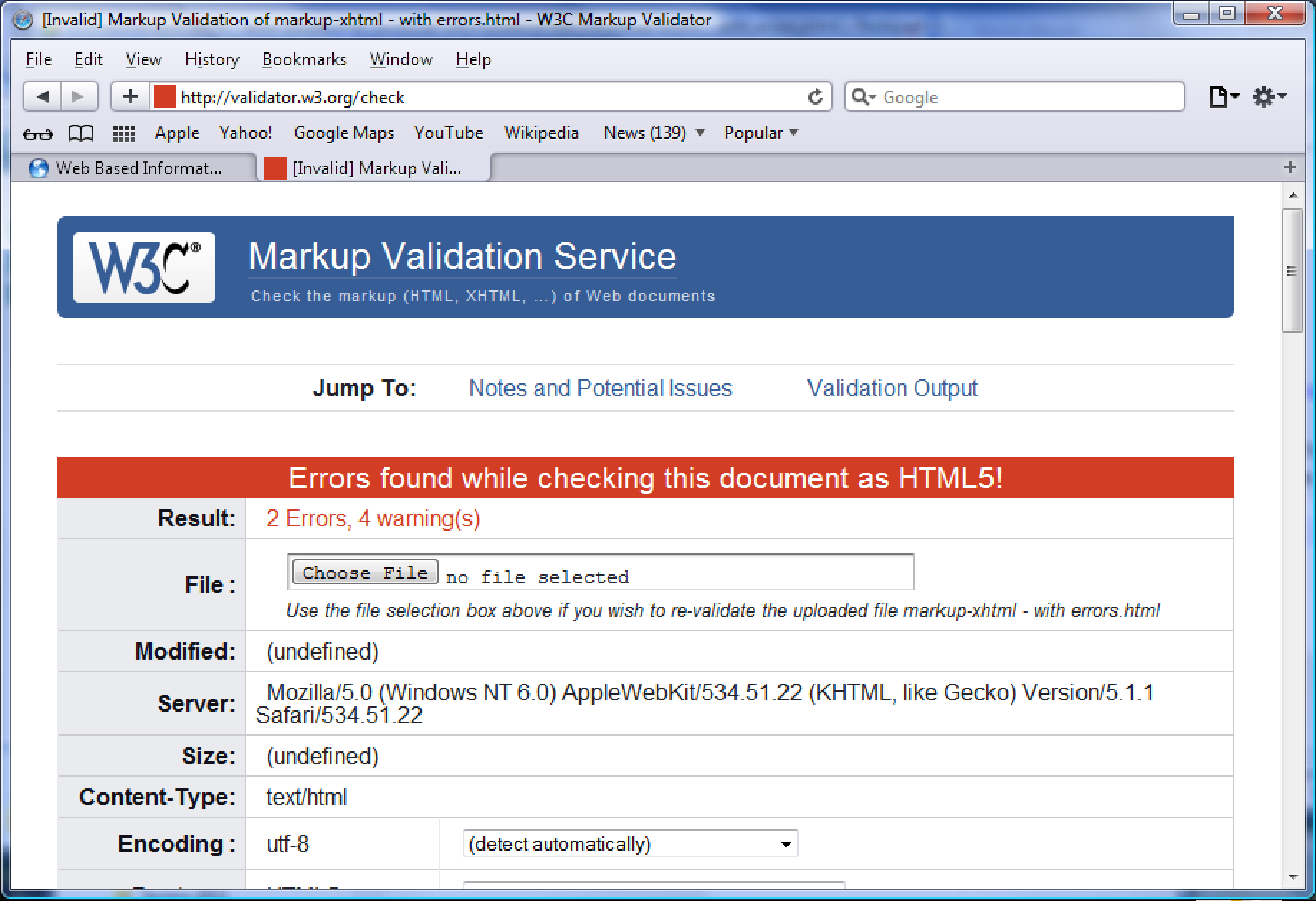
HTML5 Code With Errors – Can You Find Them?


C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\markup-xhtml - with errors.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
CH08_SaleCo2.SQL
schedule.dat
utlxplan.sql
QRYOPTDATA.SQL
NoSolutions.txt
markup-xhtml - with CSS.html
markup-xhtml - with errors.html
Hyper Text Markup Language file
length : 755 lines : 22
Ln : 3 Col : 48 Sel : 0
Dos\Windows
ANSI
INS

```
<!DOCTYPE html>
<head>
   <title>Web Based Information Technology Fall 2012
</head>
<body>
    <strong>Course Name: </strong> Web Based Information Technology<br />
    <strong>Course Number: </strong> CIS 4004 <br />
    <strong>Instructor: </strong> Dr. Mark Llewellyn <br />
    <strong>Class Meets: </strong> Tuesday and Thursday, 12:00-1:15pm, HEC 118 <br />
    <p></p>
    <strong>Course Description: </strong> Digital libraries, Media formats, Compression, Strea
        Mobile Internet and WML, Emerging technologies.  Capacity planning for web services.
    <p></p>
    <strong attribute="yes">Prerequisites: </strong>
    <ul>
       <li> CNT 3004 and,</li>
       <li> COP 3330</li>
    </ul>
</body>
</html>

```

Missing end tag </title>

No attribute named “attribute”


[Invalid] Markup Validation of markup-xhtml - with errors.html - W3C Markup Validator
File Edit View History Bookmarks Window Help
http://validator.w3.org/check
Google
Apple Yahoo! Google Maps YouTube Wikipedia News (139) Popular
Web Based Informat...
[Invalid] Markup Vali...
W3C®

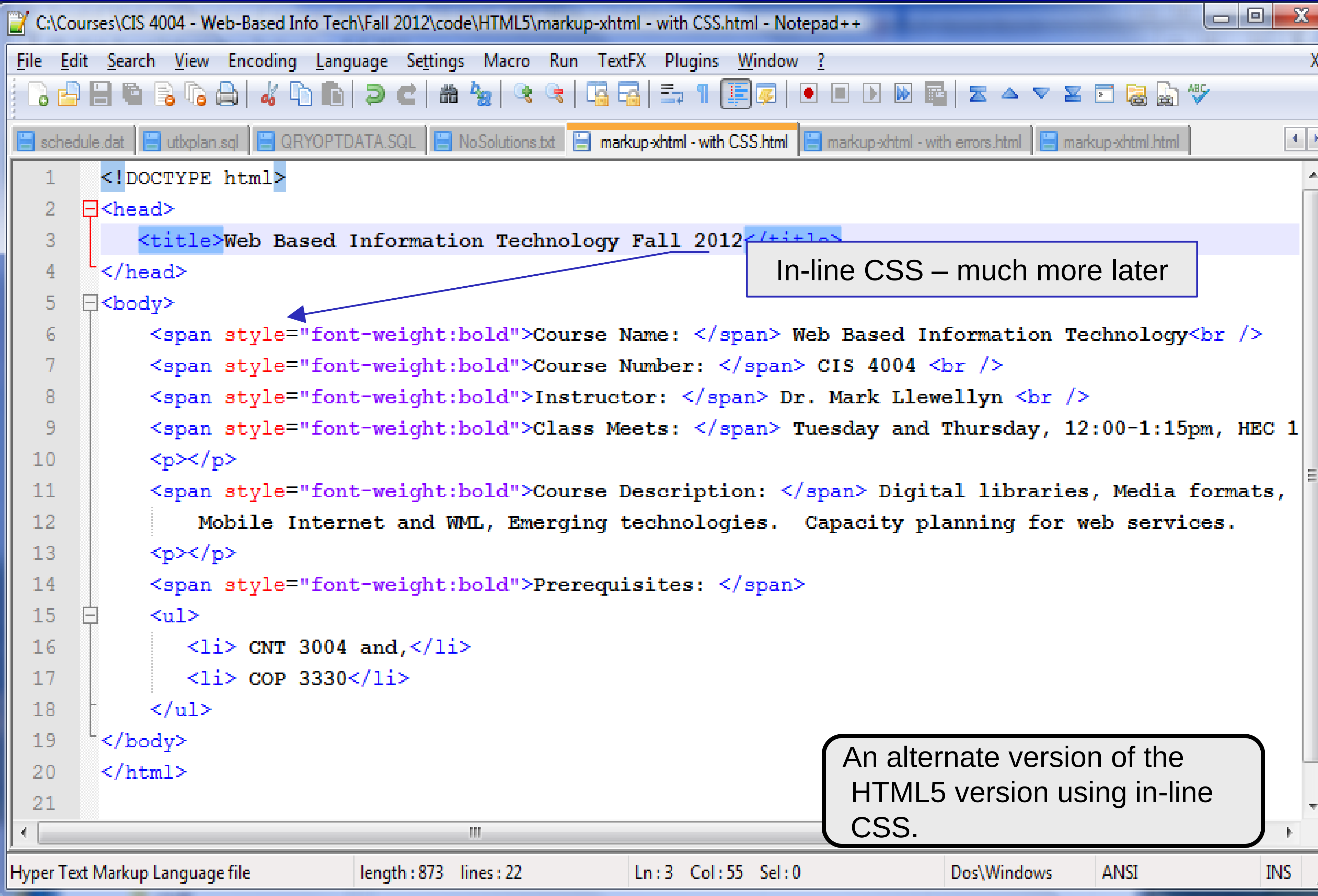
Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Jump To: Notes and Potential Issues Validation Output

Errors found while checking this document as HTML5!

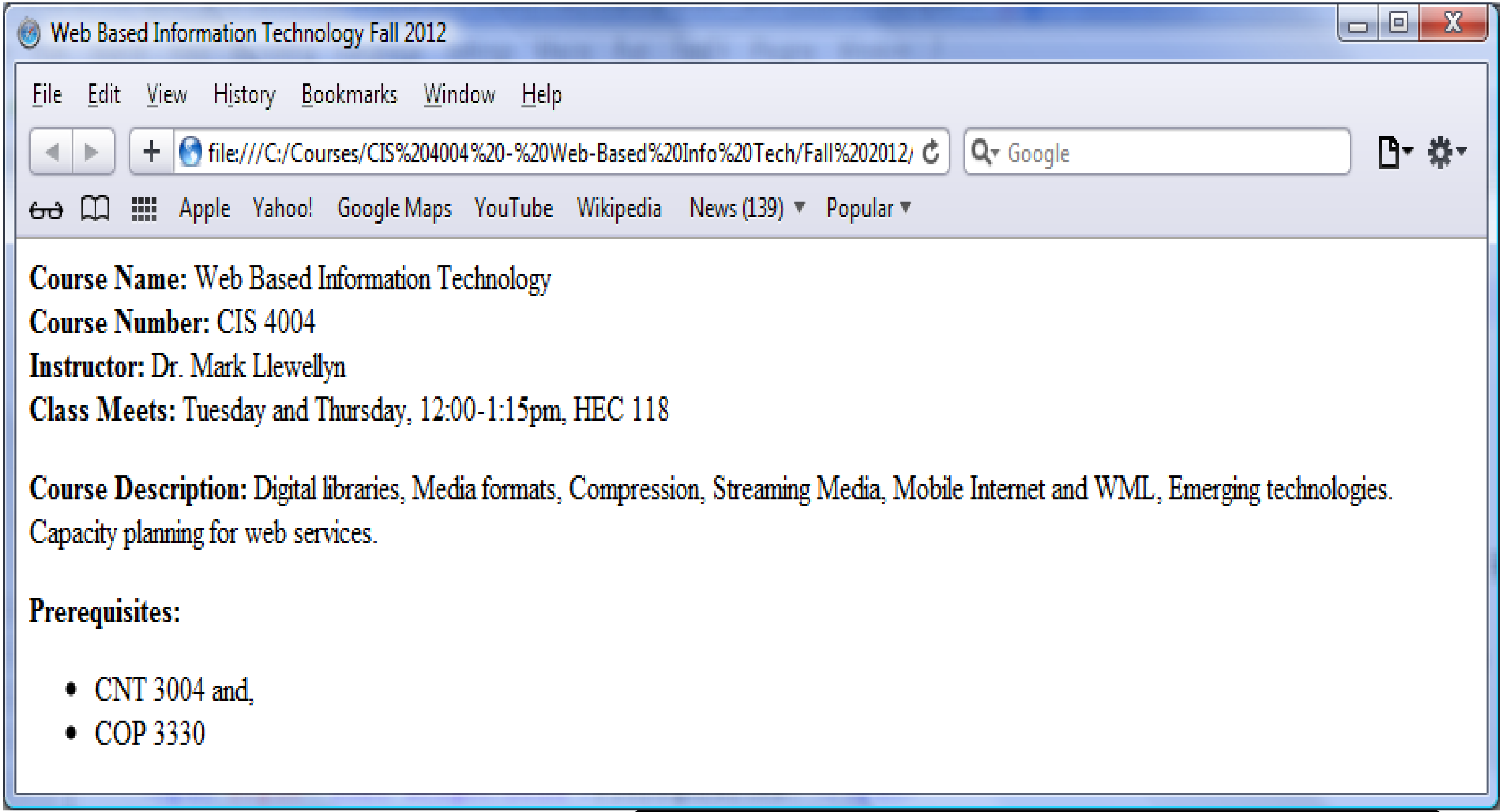
Result:	2 Errors, 4 warning(s)	
File :	Choose File no file selected *Use the file selection box above if you wish to re-validate the uploaded file markup-xhtml - with errors.html*	
Modified:	(undefined)	
Server:	Mozilla/5.0 (Windows NT 6.0) AppleWebKit/534.51.22 (KHTML, like Gecko) Version/5.1.1 Safari/534.51.22	
Size:	(undefined)	
Content-Type:	text/html	
Encoding :	utf-8	(detect automatically)


C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\markup-xhtml - with CSS.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
schedule.dat
utlxplan.sql
QRYOPTDATA.SQL
NoSolutions.txt
markup-xhtml - with CSS.html
markup-xhtml - with errors.html
markup-xhtml.html
Hyper Text Markup Language file
length : 873 lines : 22
Ln : 3 Col : 55 Sel : 0
Dos\Windows
ANSI
INS

```
<!DOCTYPE html>
<head>
   <title>Web Based Information Technology Fall 2012</title>
</head>
<body>
    <span style="font-weight:bold">Course Name: </span> Web Based Information Technology<br />
    <span style="font-weight:bold">Course Number: </span> CIS 4004 <br />
    <span style="font-weight:bold">Instructor: </span> Dr. Mark Llewellyn <br />
    <span style="font-weight:bold">Class Meets: </span> Tuesday and Thursday, 12:00-1:15pm, HEC 1
    <p></p>
    <span style="font-weight:bold">Course Description: </span> Digital libraries, Media formats,
        Mobile Internet and WML, Emerging technologies.  Capacity planning for web services.
    <p></p>
    <span style="font-weight:bold">Prerequisites: </span>
    <ul>
       <li> CNT 3004 and,</li>
       <li> COP 3330</li>
    </ul>
</body>
</html>
```

In-line CSS – much more later

An alternate version of the HTML5 version using in-line CSS.



Web Based Information Technology Fall 2012
File Edit View History Bookmarks Window Help
file:///C:/Courses/CIS%204004%20-%20Web-Based%20Info%20Tech/Fall%202012/
Google
Apple Yahoo! Google Maps YouTube Wikipedia News (139) Popular

Course Name: Web Based Information Technology
Course Number: CIS 4004
Instructor: Dr. Mark Llewellyn
Class Meets: Tuesday and Thursday, 12:00-1:15pm, HEC 118

Course Description: Digital libraries, Media formats, Compression, Streaming Media, Mobile Internet and WML, Emerging technologies. Capacity planning for web services.

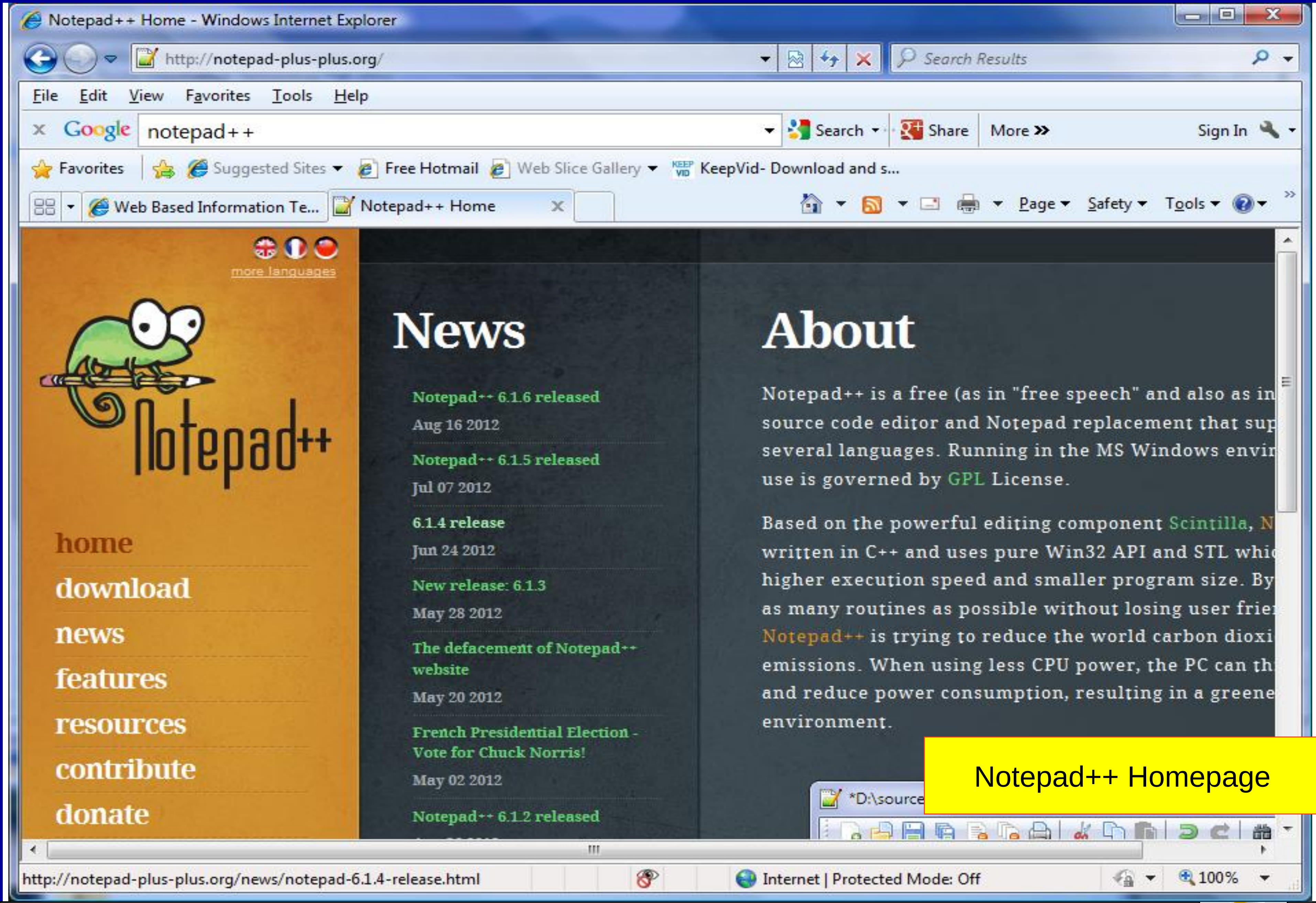
Prerequisites:

- CNT 3004 and,
- COP 3330

Safari rendering of the HTML5 markup with CSS

A Few Words On Text Editors

- Creating an XHTML document can be done in any text editor, such as Notepad.
- If you must use a word processing program, be sure that it allows you to enter text only documents and be sure that you set the program for this mode when entering XHTML documents. You do not want hidden word processing characters to appear in your XHTML documents because most browsers will not be able to correctly interpret these codes. This will lead to questionable display characteristics.
- Notepad++ is a good text editor that was developed to be syntax friendly for many different programming/markup languages including XHTML. You can download Notepad++ at: http://notepad-plus.sourceforge.net/uk/site.htm.
- The W3 organization has also developed a web-friendly editor called Amaya. The current stable version of Amaya is 11.3.1. You can download Amaya at www.w3c.org/Amaya. The Amaya environment is a graphical one that may take a bit of getting used to and many of its features will not become apparent until you begin to work on more complex XHTML documents. We'll revisit Amaya later in the semester.


Notepad++ Home - Windows Internet Explorer
http://notepad-plus-plus.org/
Search Results
File Edit View Favorites Tools Help
Google notepad++
Search Share More »
Sign In
Favorites
Suggested Sites
Free Hotmail
Web Slice Gallery
KeepVid- Download and s...
Web Based Information Te...
Notepad++ Home
Page Safety Tools
more languages
Notepad++
home
download
news
features
resources
contribute
donate
*D:\source
http://notepad-plus-plus.org/news/notepad-6.1.4-release.html
Internet | Protected Mode: Off
100%

News

Notepad++ 6.1.6 released
Aug 16 2012

Notepad++ 6.1.5 released
Jul 07 2012

6.1.4 release
Jun 24 2012

New release: 6.1.3
May 28 2012

The defacement of Notepad++ website
May 20 2012

French Presidential Election - Vote for Chuck Norris!
May 02 2012

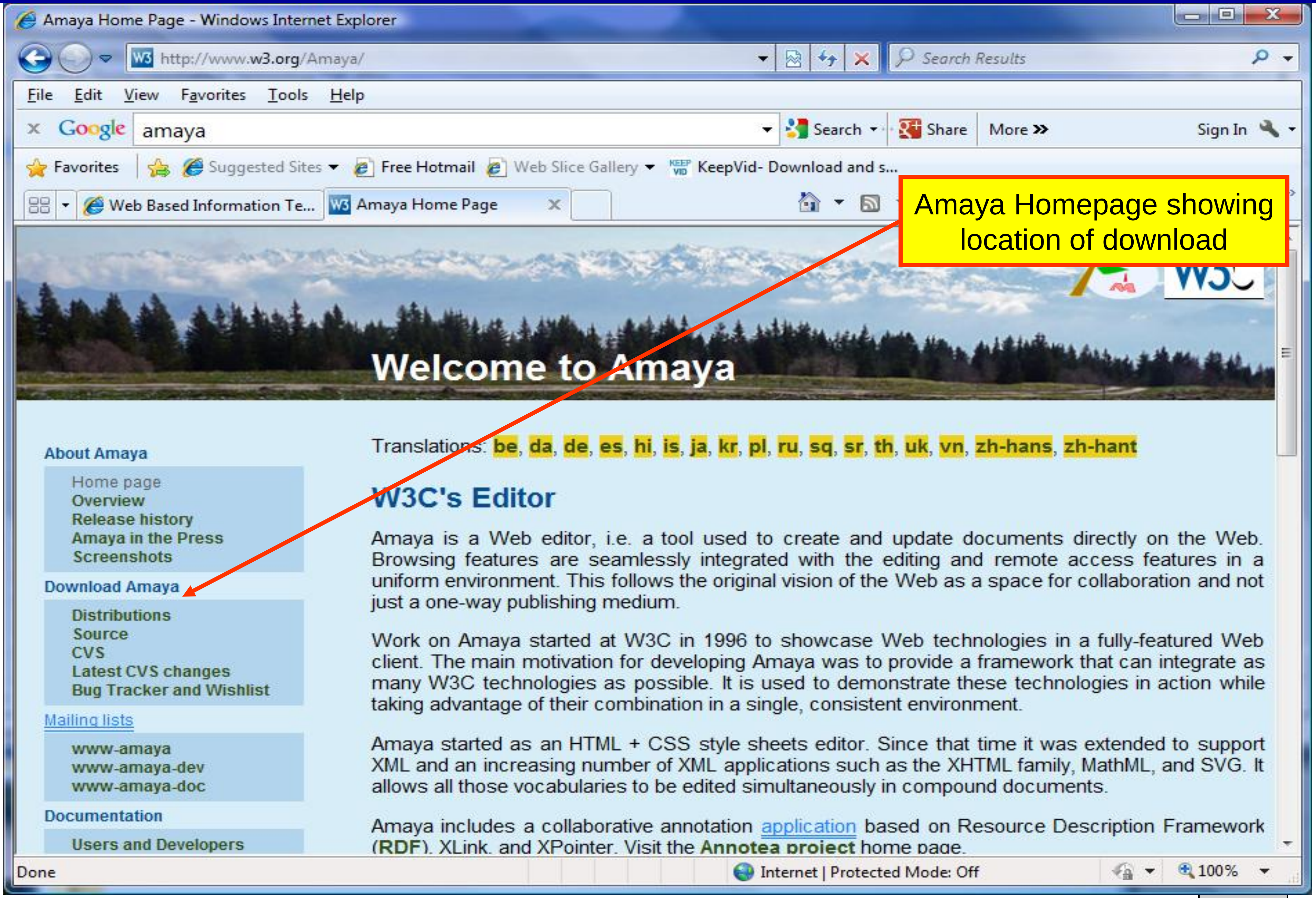
Notepad++ 6.1.2 released

About

Notepad++ is a free (as in "free speech" and also as in source code editor and Notepad replacement that sup several languages. Running in the MS Windows envir use is governed by GPL License.

Based on the powerful editing component Scintilla, N written in C++ and uses pure Win32 API and STL whic higher execution speed and smaller program size. By as many routines as possible without losing user frie Notepad++ is trying to reduce the world carbon dioxi emissions. When using less CPU power, the PC can th and reduce power consumption, resulting in a greene environment.

Notepad++ Homepage



Amaya Home Page - Windows Internet Explorer
http://www.w3.org/Amaya/
Search Results
File Edit View Favorites Tools Help
Google amaya
Search
Share
More »
Sign In
Favorites
Suggested Sites
Free Hotmail
Web Slice Gallery
KeepVid- Download and s...
Web Based Information Te...
Amaya Home Page
Done
Internet | Protected Mode: Off
100%

Amaya Homepage showing location of download

Welcome to Amaya

About Amaya
- Home page
- Overview
- Release history
- Amaya in the Press
- Screenshots

Download Amaya
- Distributions
- Source
- CVS
- Latest CVS changes
- Bug Tracker and Wishlist

Mailing lists
- www-amaya
- www-amaya-dev
- www-amaya-doc

Documentation
- Users and Developers

Translations: be, da, de, es, hi, is, ja, kr, pl, ru, sq, sr, th, uk, vn, zh-hans, zh-hant

W3C's Editor

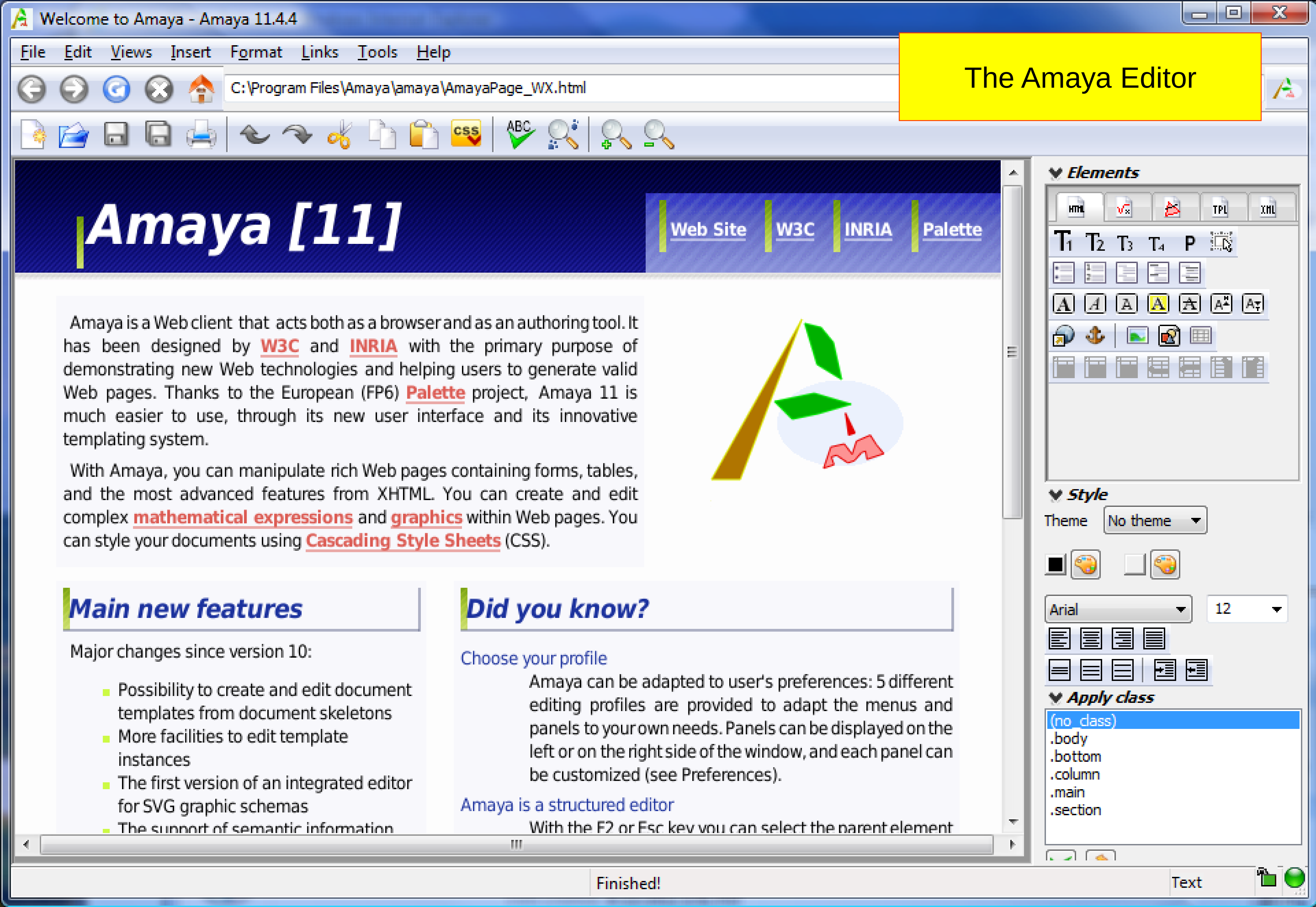
Amaya is a Web editor, i.e. a tool used to create and update documents directly on the Web. Browsing features are seamlessly integrated with the editing and remote access features in a uniform environment. This follows the original vision of the Web as a space for collaboration and not just a one-way publishing medium.

Work on Amaya started at W3C in 1996 to showcase Web technologies in a fully-featured Web client. The main motivation for developing Amaya was to provide a framework that can integrate as many W3C technologies as possible. It is used to demonstrate these technologies in action while taking advantage of their combination in a single, consistent environment.

Amaya started as an HTML + CSS style sheets editor. Since that time it was extended to support XML and an increasing number of XML applications such as the XHTML family, MathML, and SVG. It allows all those vocabularies to be edited simultaneously in compound documents.

Amaya includes a collaborative annotation application based on Resource Description Framework (RDF), XLink, and XPointer. Visit the Annotea project home page.

The Amaya Editor



C:\Program Files\Amaya\amaya\AmayaPage_WX.html

Amaya [11]

Amaya is a Web client that acts both as a browser and as an authoring tool. It has been designed by W3C and INRIA with the primary purpose of demonstrating new Web technologies and helping users to generate valid Web pages. Thanks to the European (FP6) Palette project, Amaya 11 is much easier to use, through its new user interface and its innovative templating system.

With Amaya, you can manipulate rich Web pages containing forms, tables, and the most advanced features from XHTML. You can create and edit complex mathematical expressions and graphics within Web pages. You can style your documents using Cascading Style Sheets (CSS).

Main new features

Major changes since version 10:

- Possibility to create and edit document templates from document skeletons
- More facilities to edit template instances
- The first version of an integrated editor for SVG graphic schemas
- The support of semantic information

Did you know?

Choose your profile

Amaya can be adapted to user's preferences: 5 different editing profiles are provided to adapt the menus and panels to your own needs. Panels can be displayed on the left or on the right side of the window, and each panel can be customized (see Preferences).

Amaya is a structured editor

With the F2 or Esc key you can select the parent element

Theme: No theme

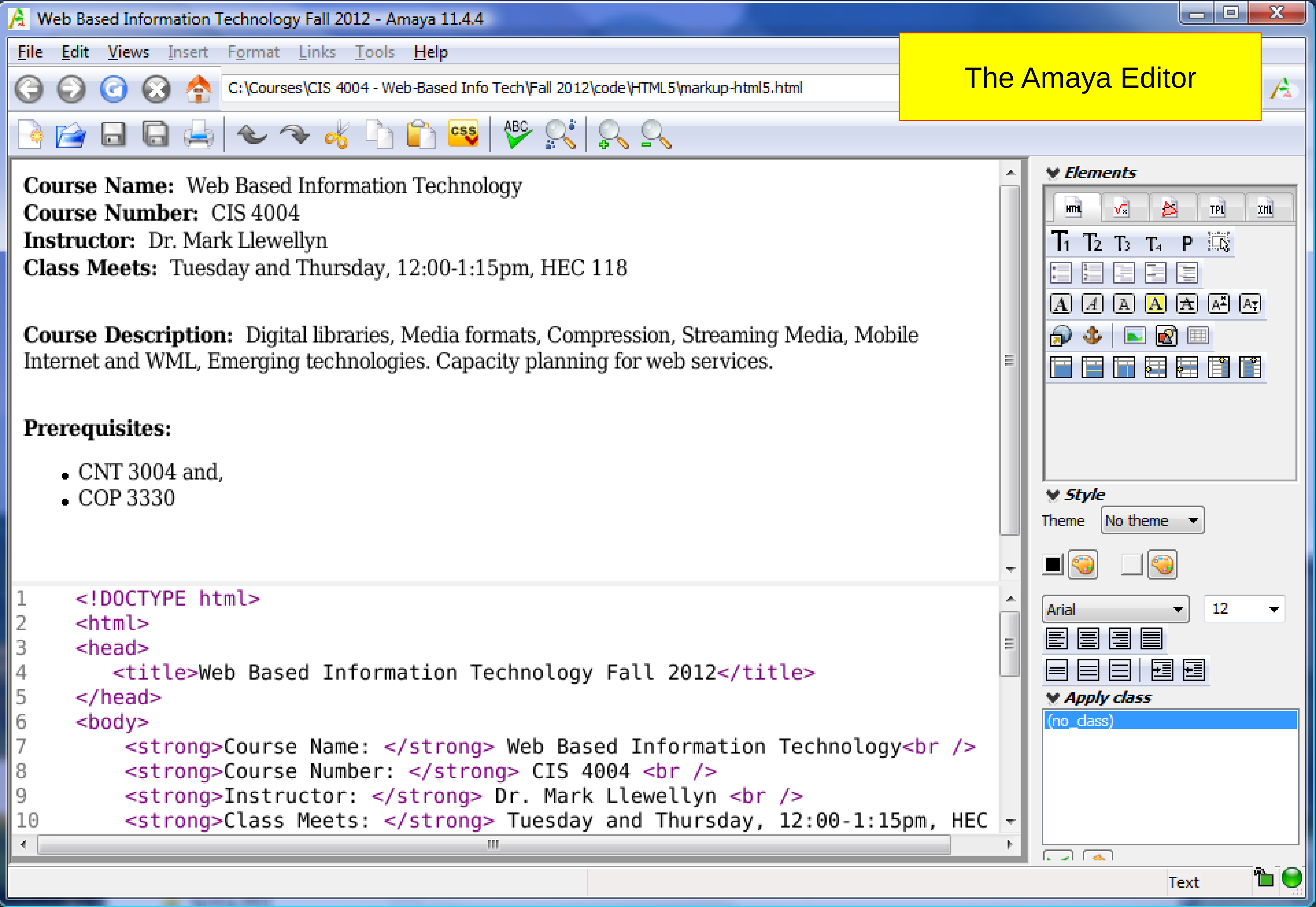
Arial 12

(no_class)
.body
.bottom
.column
.main
.section

Finished!

Text

The Amaya Editor


Web Based Information Technology Fall 2012 - Amaya 11.4.4
File
Edit
Views
Insert
Format
Links
Tools
Help
C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\markup-html5.html
Elements
Style
Apply class
Text

Theme: No theme

Arial 12

(no_class)

Course Name: Web Based Information Technology
Course Number: CIS 4004
Instructor: Dr. Mark Llewellyn
Class Meets: Tuesday and Thursday, 12:00-1:15pm, HEC 118

Course Description: Digital libraries, Media formats, Compression, Streaming Media, Mobile Internet and WML, Emerging technologies. Capacity planning for web services.

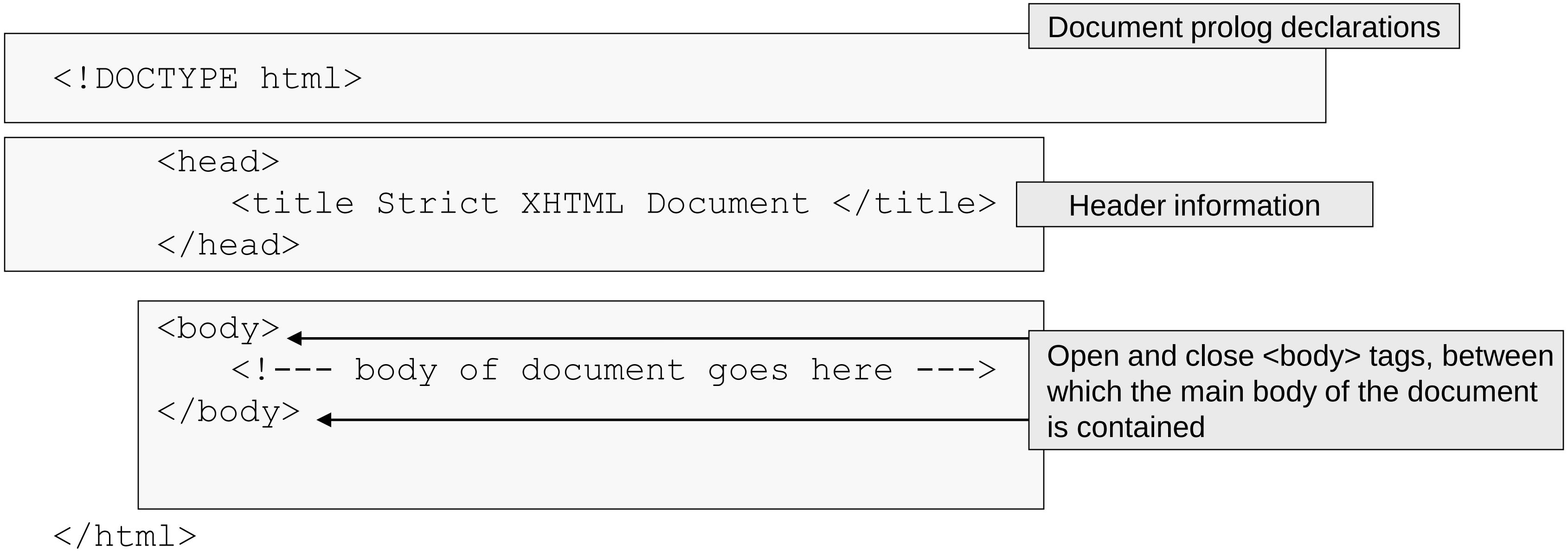
Prerequisites:

- CNT 3004 and,
- COP 3330

```
<!DOCTYPE html>
<html>
<head>
   <title>Web Based Information Technology Fall 2012</title>
</head>
<body>
    <strong>Course Name: </strong> Web Based Information Technology<br />
    <strong>Course Number: </strong> CIS 4004 <br />
    <strong>Instructor: </strong> Dr. Mark Llewellyn <br />
    <strong>Class Meets: </strong> Tuesday and Thursday, 12:00-1:15pm, HEC
```

How HTML5 Documents Are Structured

- XHTML documents are comprised of a simple three-part framework:
  1. Document prolog
  2. Header section
  3. Body of document.



```
<!DOCTYPE html>

      <head>
          <title Strict XHTML Document </title>
      </head>

      <body>
          <!--- body of document goes here --->
      </body>

</html>
```

Document prolog declarations

Header information

Open and close <body> tags, between which the main body of the document is contained

Document Framework Elements

- The elements that make up the framework of HTML5 documents do not produce any output in a browser window. Instead, they provide information to the program about the document.

- The elements that makeup the framework are: `<html>`, `<head>`, `<title>`, and `<body>`. We'll look at each of these more closely.

The `<html>` Element

- The `<html>` element is the root element of an HTML5 document, within which every other element in the document is contained (recall our discussion on nesting of elements).
- The document begins with the `<html>` start tag and ends with the `</html>` end tag. The header and body information of the document are contained in the root element.

The <head> Element

- The start <head> tag comes directly after the <html> start tag in an HTML5 document. This element must be placed inside the <html> element.

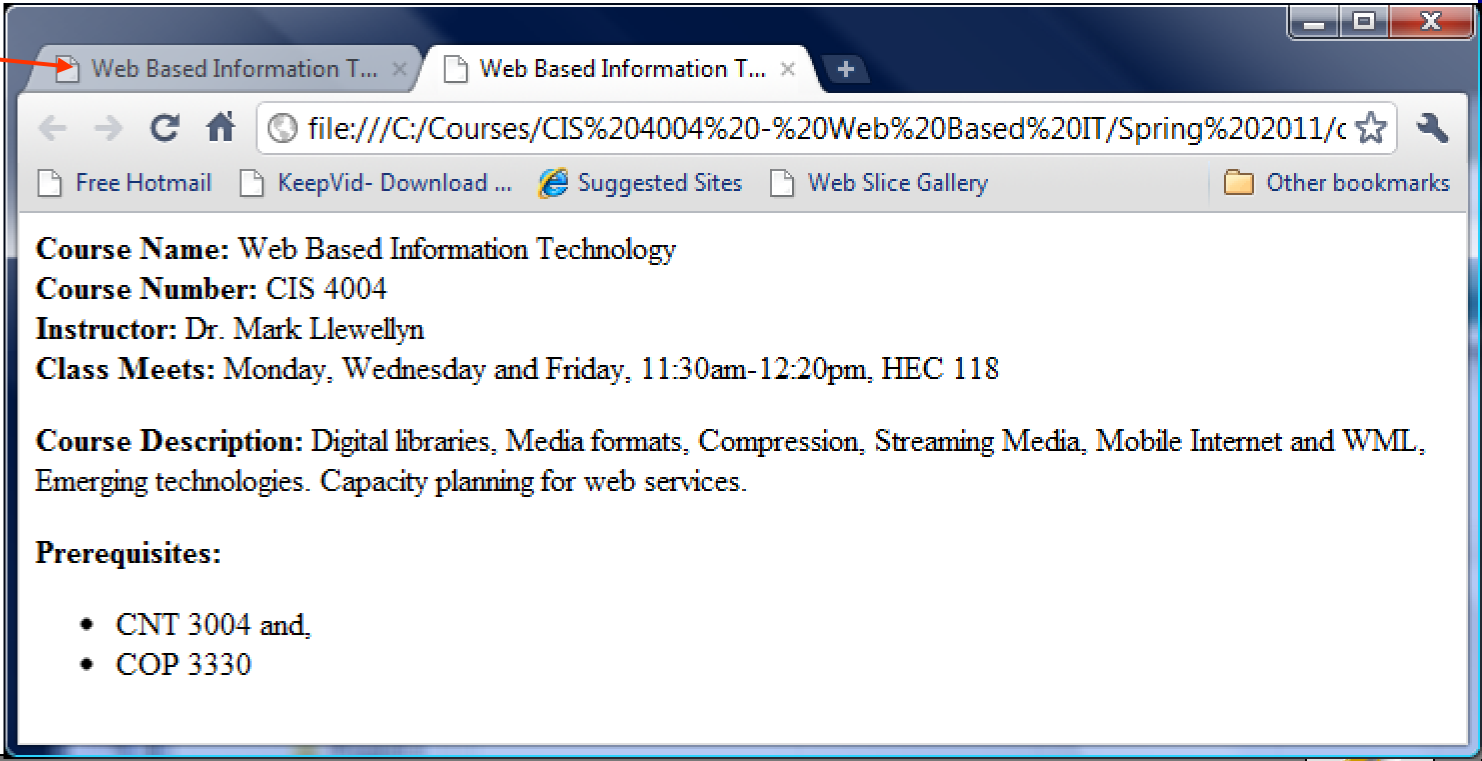
- It contains information about the document that is mainly used by programs, such as keywords for search engines and link information that defines the relationship this document has to other documents.

- The <head> tag also contains the required <title> element. The following is a list of elements that can be contained in the <head> element – all but the <title> element are optional.
  - <title> - defines the title of the document.
  - <base> - defines the document base URL, which is used for relative links in the document.
  - <link> - defines the relationship of this document to other documents.
  - <meta> - defines additional information about the document, including the document's content type and special instructions for browsers and search engines.
  - <script> - defines links to scripts used within the document, such as Javascript.
  - <style> - defines links to style sheets to be used within the document, such as CSS.

The `<title>` Element

- The `<title>` element is the only required element within the `<head>` element. It must be contained within the open and close tags of the `<head>` element.
- There can only be one `<title>` element per document. It defines the title of the document that is displayed in the title bar of the browser window as well as the name of bookmarks to that page.
- Most search engines use the content of the `<title>` element as the text to display on their results pages as well.

title


Web Based Information T...
Web Based Information T...
file:///C:/Courses/CIS%204004%20-%20Web%20Based%20IT/Spring%202011/c
Free Hotmail
KeepVid- Download ...
Suggested Sites
Web Slice Gallery
Other bookmarks

Course Name: Web Based Information Technology
Course Number: CIS 4004
Instructor: Dr. Mark Llewellyn
Class Meets: Monday, Wednesday and Friday, 11:30am-12:20pm, HEC 118

Course Description: Digital libraries, Media formats, Compression, Streaming Media, Mobile Internet and WML, Emerging technologies. Capacity planning for web services.

Prerequisites:

- CNT 3004 and,
- COP 3330

The `<body>` Element

- The `<body>` element contains the content and all of the markup elements of the document.
- The body of the document is contained between the open `<body>` tag and the ending `</body>` tag.
- All of the other elements we will cover are contained within the `<body>` element.

Basic Formatting Elements In HTML5

- Now that you understand how HTML5 documents are structured, let’s start building some Web pages. We’ll start with basic formatting elements, show some examples of how to use each of the elements, and create a few documents to illustrate how they look in a Web browser.

- We’ll start by looking at block-level formatting elements.

Block-level Formatting Elements – Summary Chart

Element Name	Formatting Style
`<p>...</p>`	Paragraph element
`<br />`	Line break (empty element)
`<h1>...</h1>` to `<h6>...</h6>`	Heading elements (1 is largest, 6 is smallest)
`<hr />`	Horizontal rule (empty element)
`<div>...</div>`	Section divider

Block-level Formatting Elements

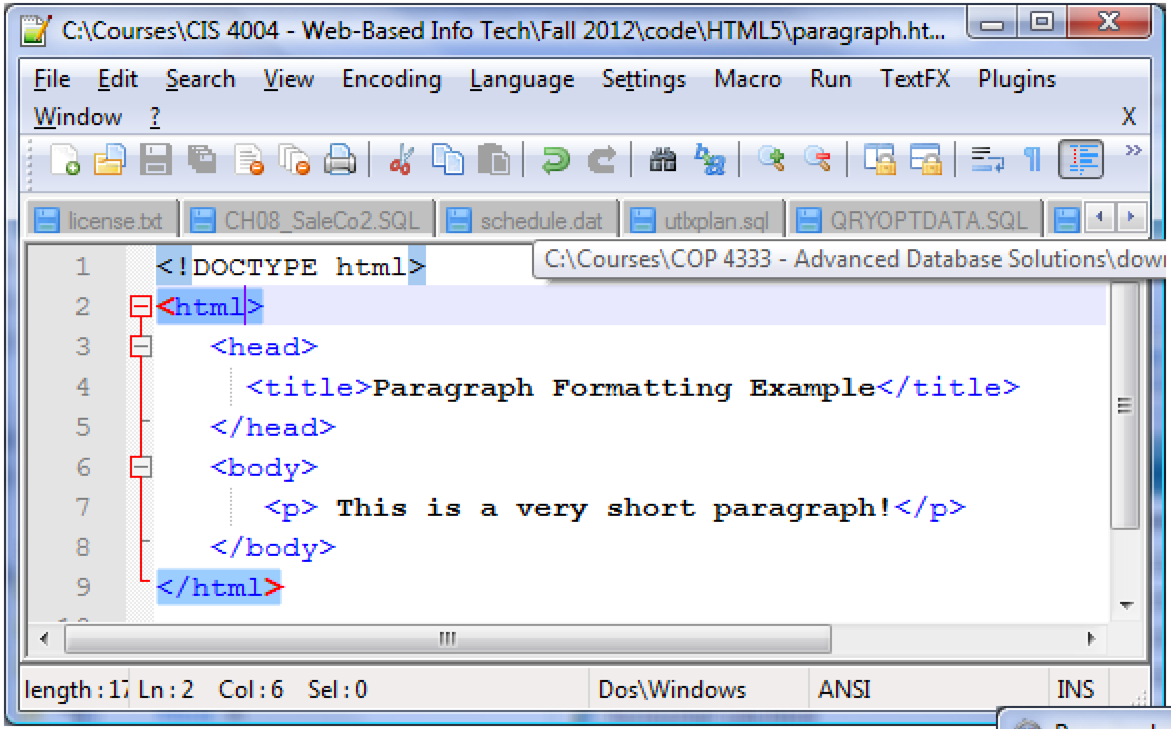
- Documents are broken into logical sections based on the document content to make it easer for users to read.
- The elements shown in the table on the previous page are used to break documents into logical chunks and to label the main content headings.
- These elements are referred to as block-level elements because they describe blocks of content.
- We'll examine each of these elements separately.

The `<p>` Element

- The `<p>` element divides content into paragraphs.
- The `<p>` tag designates the beginning of a paragraph, and the `</p>` tag ends the paragraph.
- Most browsers will automatically insert a double line return (carriage return) around the paragraph element.
- Example:

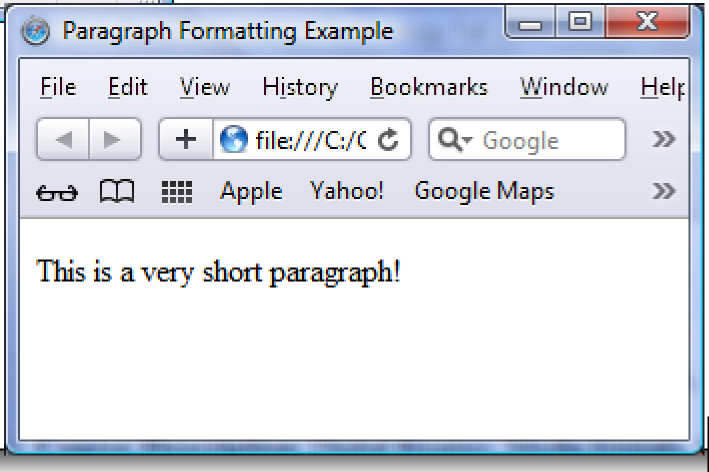
```
<p>This is a very short paragraph.</p>
```

The <p> Element


C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\paragraph.ht...
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins
Window ?
license.txt
CH08_SaleCo2.SQL
schedule.dat
utlxplan.sql
QRYOPTDATA.SQL
C:\Courses\COP 4333 - Advanced Database Solutions\dow
length : 17 Ln : 2 Col : 6 Sel : 0
Dos\Windows
ANSI
INS

```
<!DOCTYPE html>
<html>
    <head>
        <title>Paragraph Formatting Example</title>
    </head>
    <body>
        <p> This is a very short paragraph!</p>
    </body>
</html>
```

HTML5


Paragraph Formatting Example
File Edit View History Bookmarks Window Help
file:///C:/C
Google
Apple Yahoo! Google Maps

This is a very short paragraph!

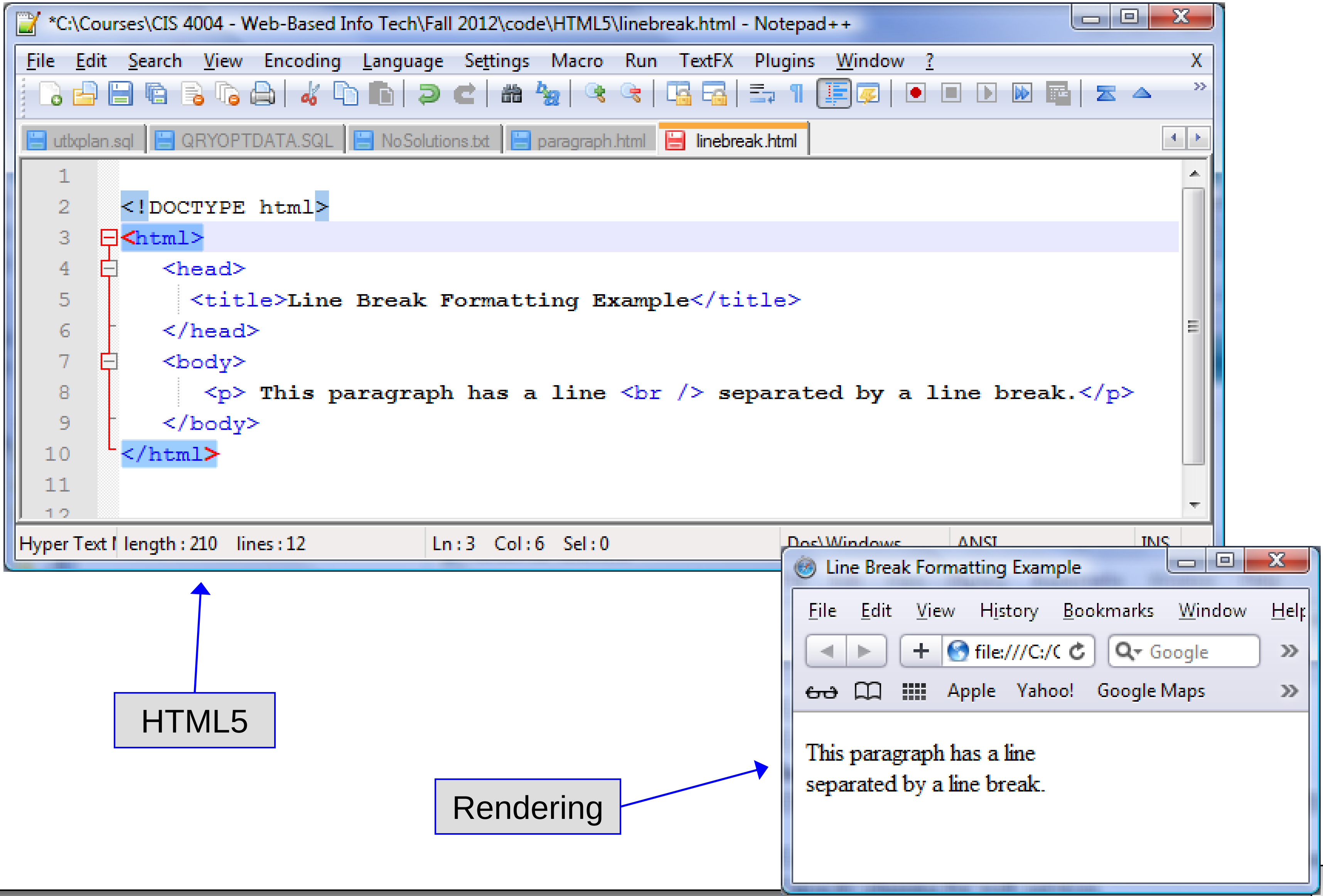
Rendering

The `<br />` Element

- The `<br />` element is the line break element. Similar to the `<p>` element, it is used to break up sections of text.
- The `<br />` element causes the browser to create a single line return (single carriage return).
- The `<br />` element is an empty element and must end with `/>` in order to conform to the rules of a well-formed document.
- Example:

```
<p>This paragraph has a line <br />
   separated by a line break.</p>
```

The `<br />` Element


*C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\linebreak.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
utlxplan.sql
QRYOPTDATA.SQL
NoSolutions.txt
paragraph.html
linebreak.html

```html
<!DOCTYPE html>
<html>
    <head>
        <title>Line Break Formatting Example</title>
    </head>
    <body>
        <p> This paragraph has a line <br /> separated by a line break.</p>
    </body>
</html>
```

Hyper Text I length : 210 lines : 12 Ln : 3 Col : 6 Sel : 0 Dos\Windows ANSI INS
Line Break Formatting Example
File Edit View History Bookmarks Window Help
file:///C:/C
Google
Apple Yahoo! Google Maps

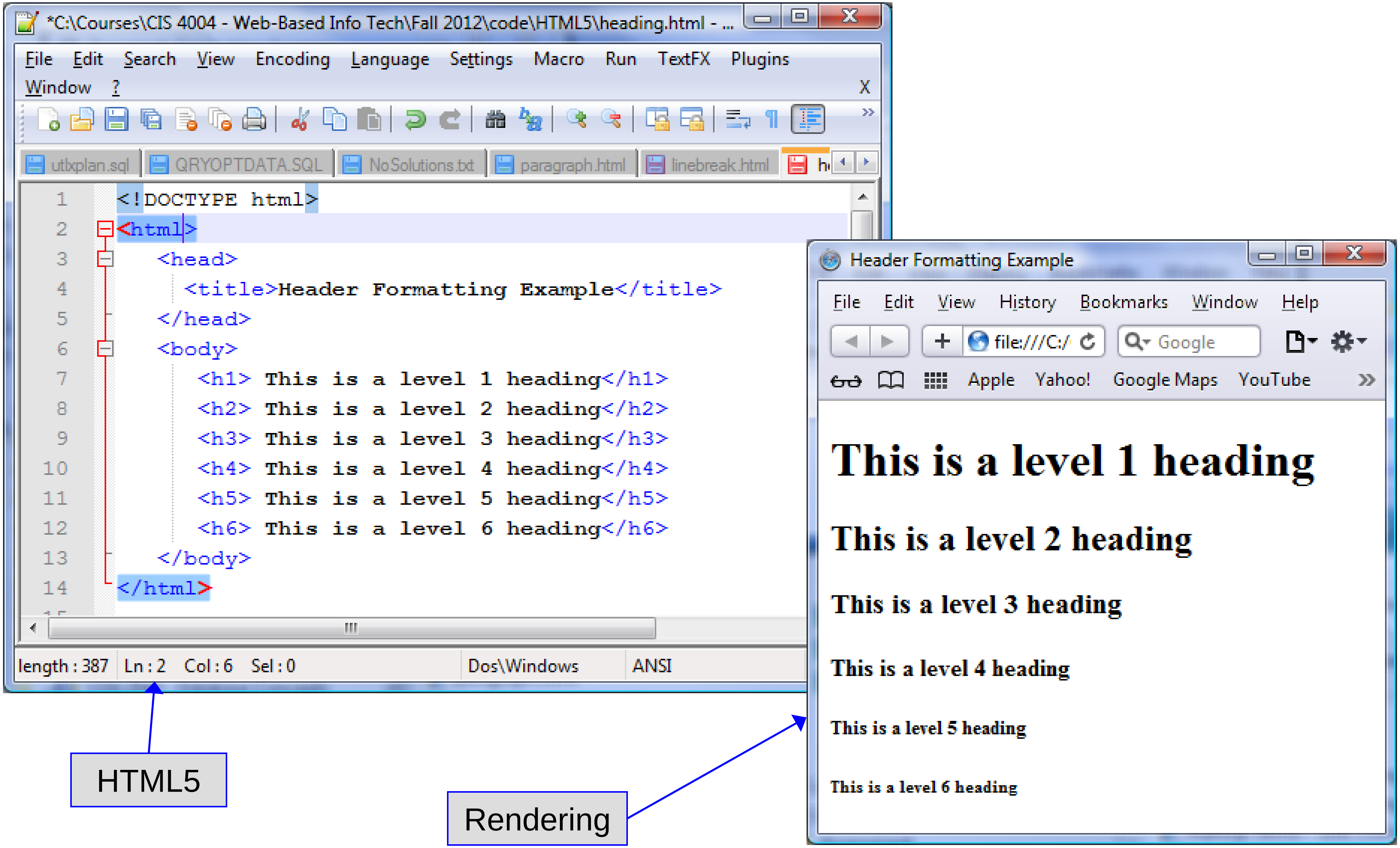
This paragraph has a line
separated by a line break.

The `<h1>` ... `<h6>` Elements

- These elements are the heading elements. They are used to label section headings of a document.
- There are six heading levels: `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, and `<h6>`.
- The `<h1>` head element should be used to label the top-most heading, and the rest of the elements should be used for subheads, much like a table of contents hierarchy.
- The browser will display the font for each of these levels differently, starting with a larger font for `<h1>` and progressively getting smaller as the heading number increases.
- Example:

```
<h1> This is a level 1 heading</h1>
<h2> This is a level 2 heading</h2>
<h3> This is a level 3 heading</h3>
<h4> This is a level 4 heading</h4>
<h5> This is a level 5 heading</h5>
<h6> This is a level 6 heading</h6>
```

The `<h1> ... <h6>` Elements



*C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\heading.html - ...

File Edit Search View Encoding Language Settings Macro Run TextFX Plugins
Window ?
utlxplan.sql
QRYOPTDATA.SQL
NoSolutions.txt
paragraph.html
linebreak.html

```html
<!DOCTYPE html>
<html>
    <head>
      <title>Header Formatting Example</title>
    </head>
    <body>
       <h1> This is a level 1 heading</h1>
       <h2> This is a level 2 heading</h2>
       <h3> This is a level 3 heading</h3>
       <h4> This is a level 4 heading</h4>
       <h5> This is a level 5 heading</h5>
       <h6> This is a level 6 heading</h6>
    </body>
</html>
```

length : 387 Ln : 2 Col : 6 Sel : 0 Dos\Windows ANSI

Header Formatting Example

File Edit View History Bookmarks Window Help
file:///C:/
Google
Apple Yahoo! Google Maps YouTube

This is a level 1 heading

This is a level 2 heading

This is a level 3 heading

This is a level 4 heading

This is a level 5 heading

This is a level 6 heading

HTML5

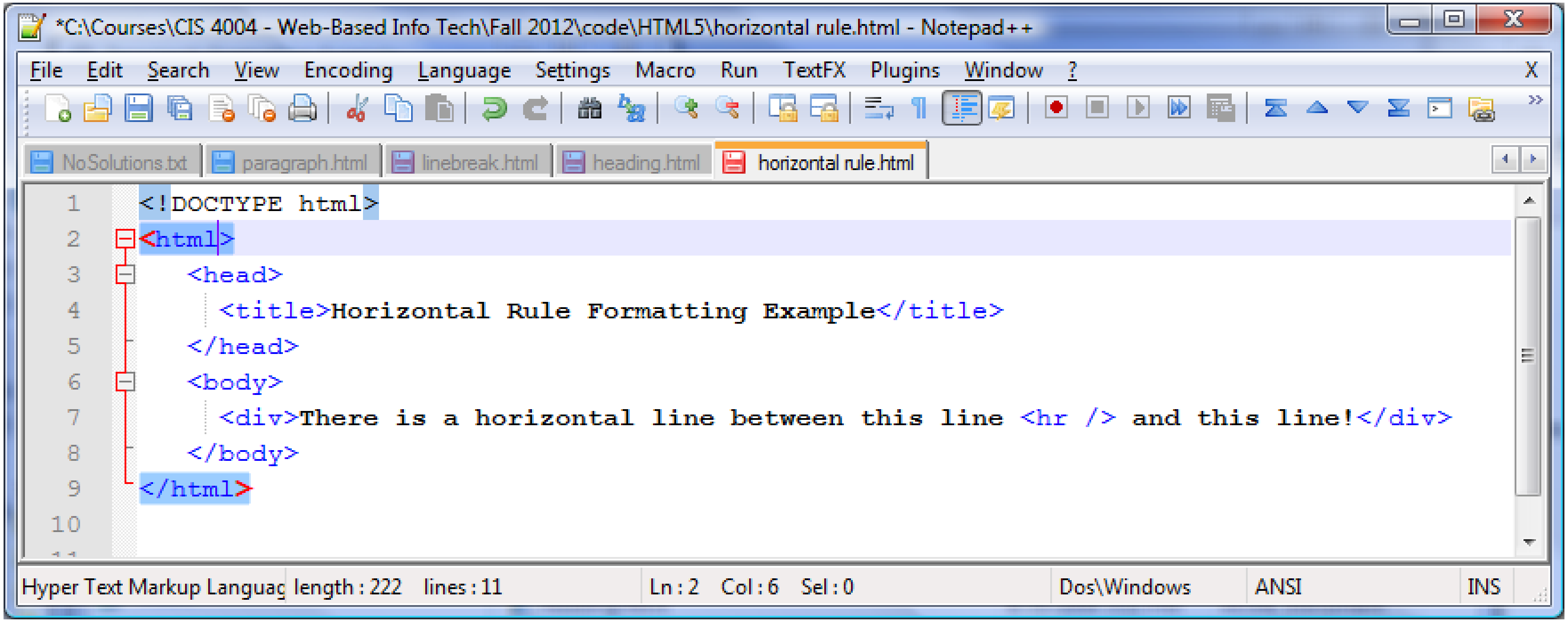
Rendering

The `<hr />` Element

- The `<hr />` element is the horizontal rule element, used to create a visible horizontal line on the Web page to indicate a section break or change in content.
- XHTML Transitional and Frameset provide a set of attributes that can be used with this element to customize the rule. In XHTML Strict customization of the line is done via CSS.
- Example:

```
There is a horizontal line between this line <hr /> and this line.
```

The `<hr />` Element



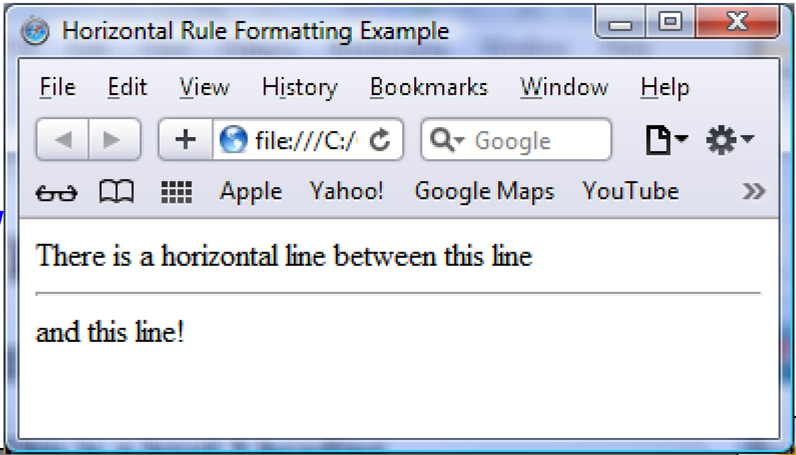
*C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\horizontal rule.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
NoSolutions.txt
paragraph.html
linebreak.html
heading.html
horizontal rule.html

```html
<!DOCTYPE html>
<html>
    <head>
      <title>Horizontal Rule Formatting Example</title>
    </head>
    <body>
      <div>There is a horizontal line between this line <hr /> and this line!</div>
    </body>
</html>
```

Hyper Text Markup Languag length : 222 lines : 11 Ln : 2 Col : 6 Sel : 0 Dos\Windows ANSI INS

HTML5

Rendering



Horizontal Rule Formatting Example
File Edit View History Bookmarks Window Help
file:///C:/
Google
Apple Yahoo! Google Maps YouTube

There is a horizontal line between this line

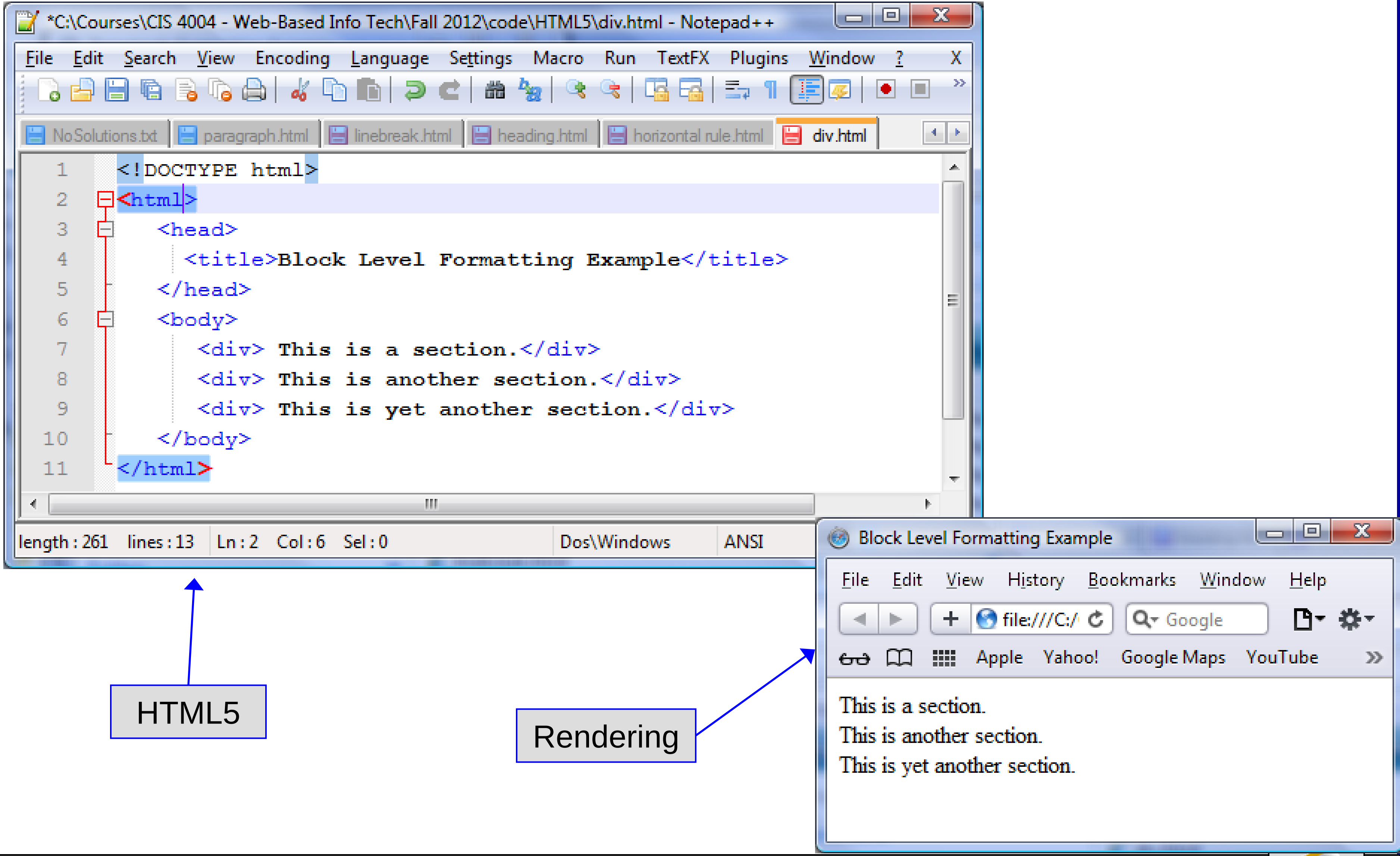
and this line!

The <div> Element

- The <div> element is used to divide sections of content. This element is used to label sections of the document, and can contain any number of other elements.
- This element can use the id and class core XHTML attributes to identify the various sections of the document to be used with parser programs.
- Example:

```
<div> This is a section.</div>

 <div> This is another section.</div>
```

The `<div>` Element



*C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\div.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
NoSolutions.txt paragraph.html linebreak.html heading.html horizontal rule.html div.html

```html
<!DOCTYPE html>
<html>
    <head>
      <title>Block Level Formatting Example</title>
    </head>
    <body>
       <div> This is a section.</div>
       <div> This is another section.</div>
       <div> This is yet another section.</div>
    </body>
</html>
```

length : 261 lines : 13 Ln : 2 Col : 6 Sel : 0 Dos\Windows ANSI

Block Level Formatting Example
File Edit View History Bookmarks Window Help
file:///C:/
Google
Apple Yahoo! Google Maps YouTube

This is a section.
This is another section.
This is yet another section.

HTML5

Rendering

Text Formatting Elements

- Text formatting elements are referred to as character-level elements because, unlike the block-level elements, which describe blocks of content, these elements describe the text itself.

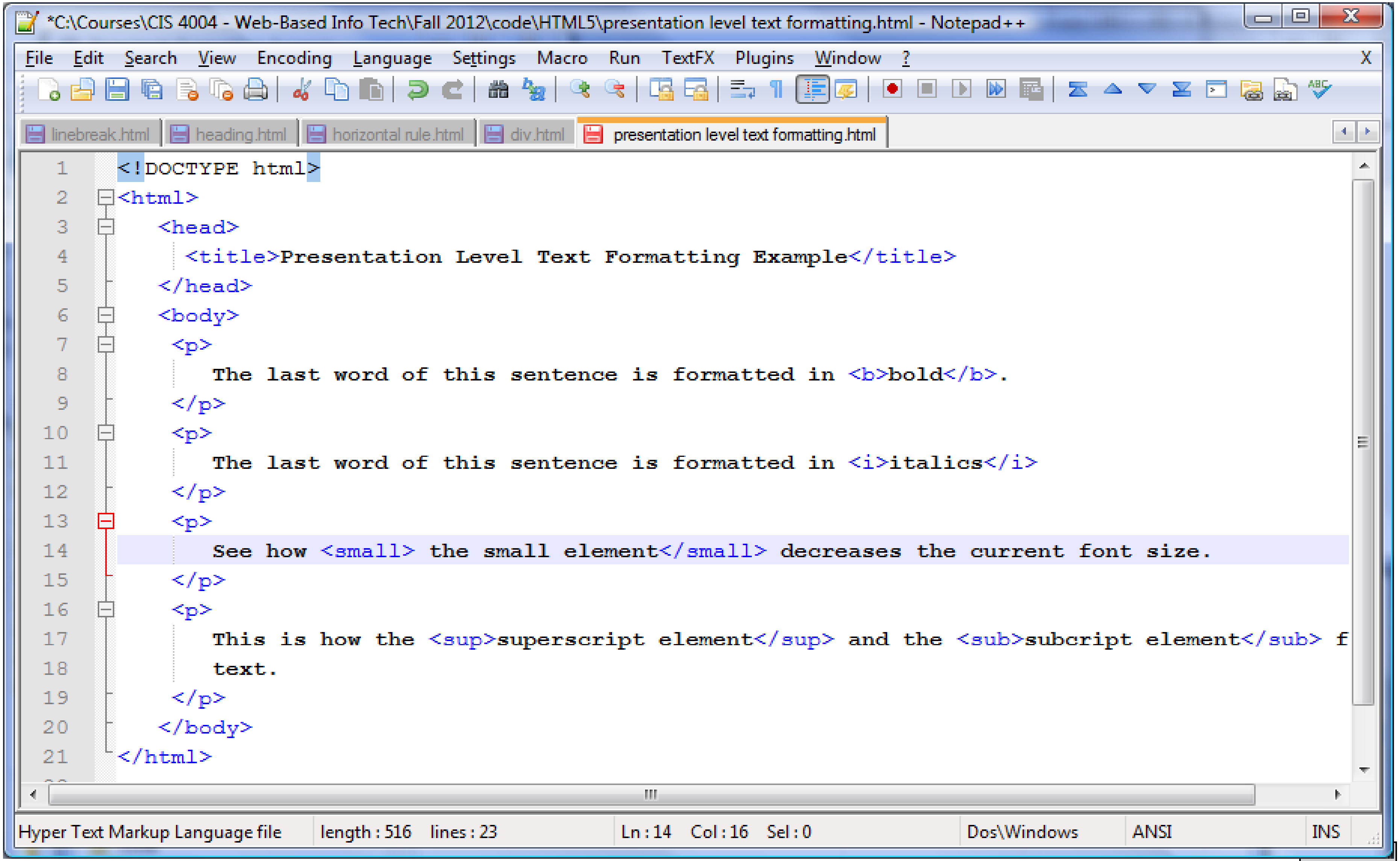
- Character-level elements describe the formatting of words or phrases as opposed to sections or paragraphs.

- There are two basic groups of text formatting elements: presentation styles, and logical styles.
  - Presentational styles describe how the text should be displayed, in bold type or italics, for example.
  - Logical styles describe the meaning of the style more than the actual format.

Presentational Text Formatting – Summary Chart

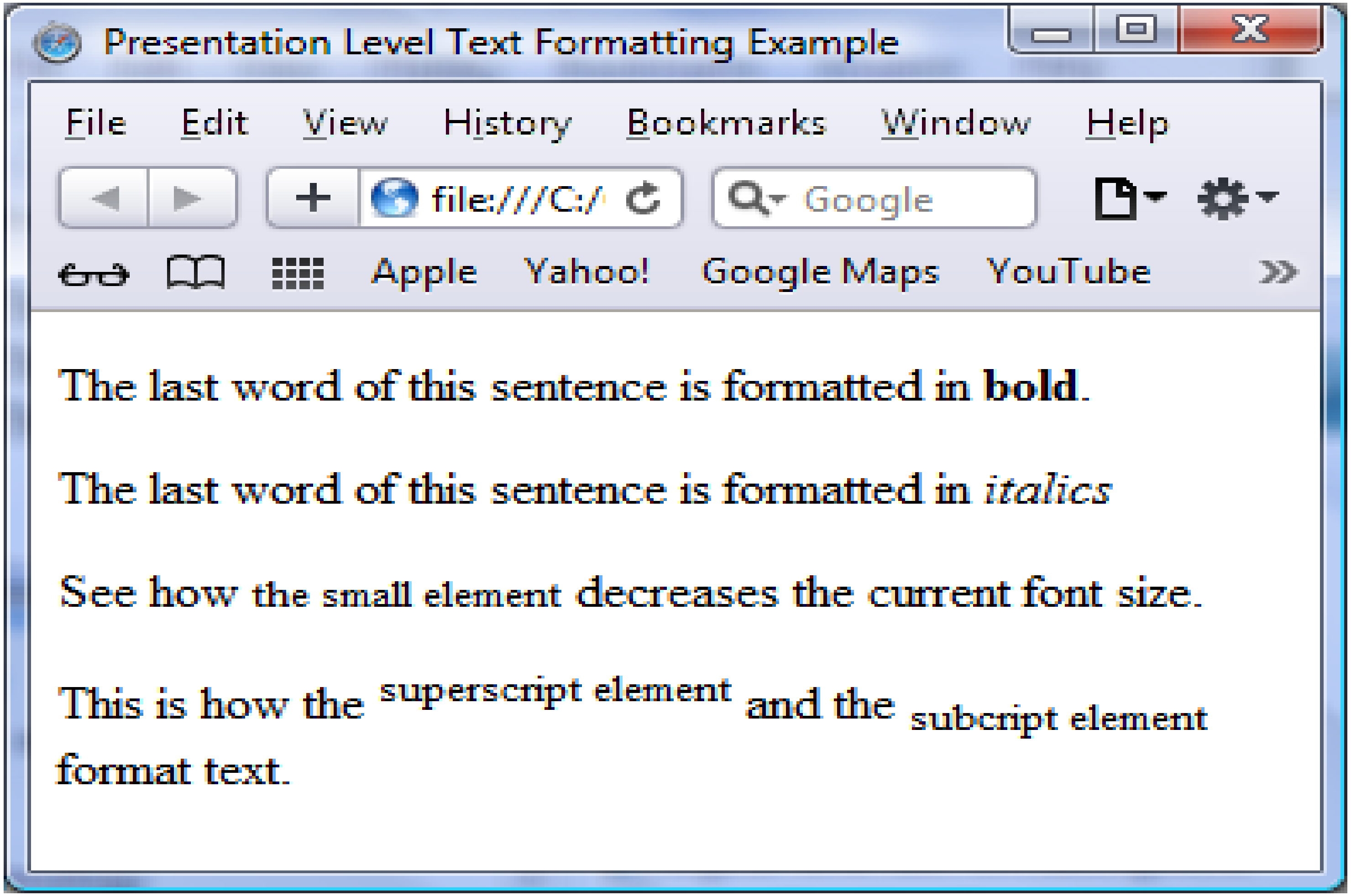
Element Name	Formatting Style
`<b>...</b>`	Bold font style
`<i>...</i>`	Italic font style
`<small>...</small>`	Decrease current font size
`<sub>...</sub>`	Subscripted text
`<sup>...</sup>`	Superscripted text

Presentational Text Formatting – Example


*C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\presentation level text formatting.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
linebreak.html
heading.html
horizontal rule.html
div.html
presentation level text formatting.html
Hyper Text Markup Language file
length : 516 lines : 23
Ln : 14 Col : 16 Sel : 0
Dos\Windows
ANSI
INS

```html
<!DOCTYPE html>
<html>
   <head>
     <title>Presentation Level Text Formatting Example</title>
   </head>
   <body>
    <p>
       The last word of this sentence is formatted in <b>bold</b>.
    </p>
    <p>
       The last word of this sentence is formatted in <i>italics</i>
    </p>
    <p>
       See how <small> the small element</small> decreases the current font size.
    </p>
    <p>
       This is how the <sup>superscript element</sup> and the <sub>subcript element</sub> f
       text.
    </p>
   </body>
</html>
```

Presentational Text Formatting – Example



The last word of this sentence is formatted in **bold**.

The last word of this sentence is formatted in *italics*

See how the small element decreases the current font size.

This is how the $^{\text{superscript element}}$ and the $_{\text{subcript element}}$ format text.

Logical Text Formatting Elements

- Logical text formatting describe the meaning of the style more than the actual format.

- Initially, browsers were left to determine the presentation of these tags as they saw fit, but over time certain standards were developed, and these are unlikely to change in the foreseeable future.

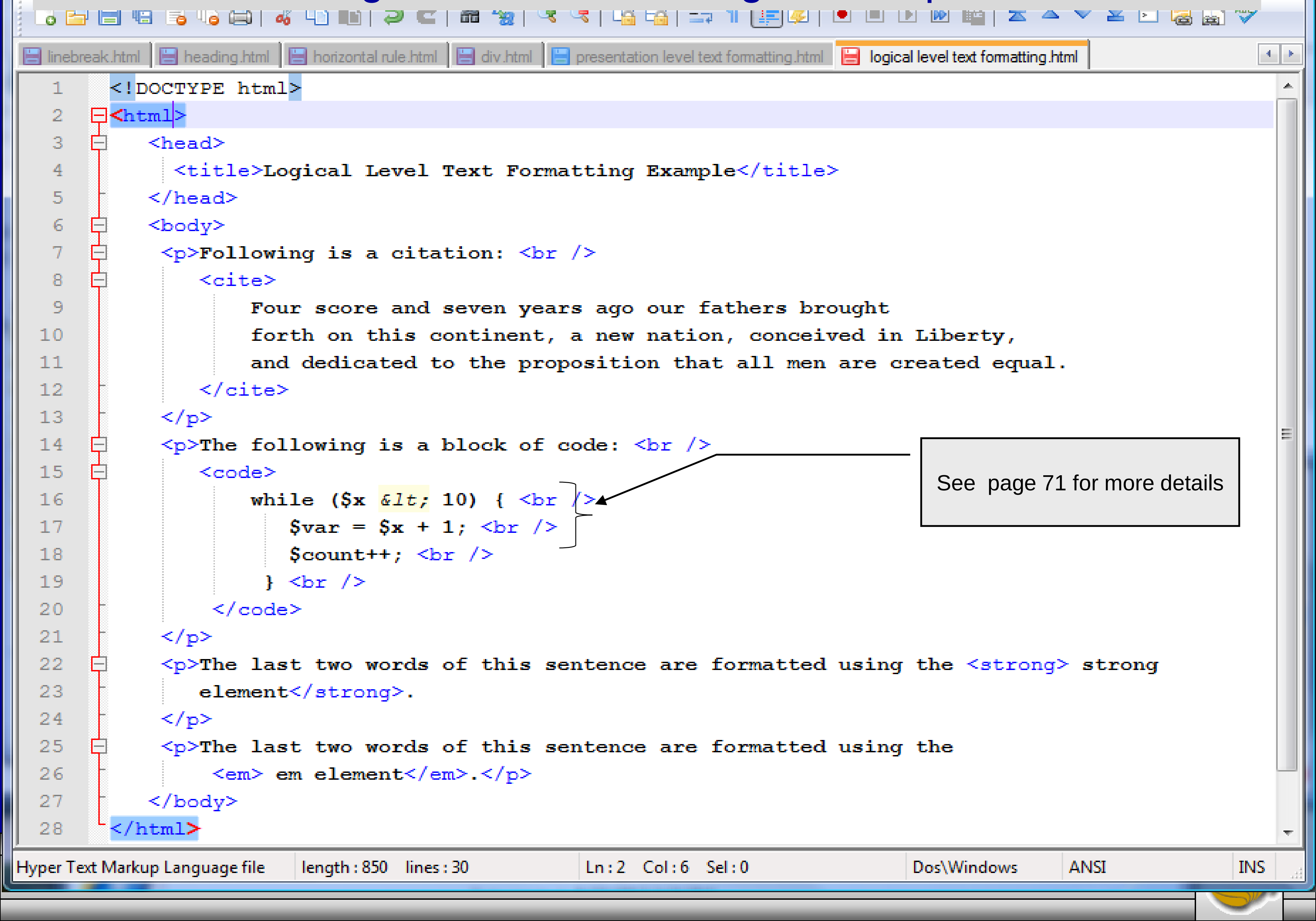
- For example, if you want a certain type, like bold, you should use the `<b>` elements, but the `<strong>` element will give you the same effect, because most browsers will interpret `<strong>` as `<b>.`

- The table on the next page lists the most commonly used logical elements.

Logical Text Formatting – Summary Chart

Element Name	Formatting Style
`<cite>...</cite>`	Defines a citation
`<code>...</code>`	Presents computer code examples
`<em>...</em>`	Emphasis. In most browsers, this is italics
`<strong>...</strong>`	Emphasis. In most browsers, this is bold
`<span>...</span>`	Provides a logical inline grouping with no predefined look.

Logical Text Formatting – Example

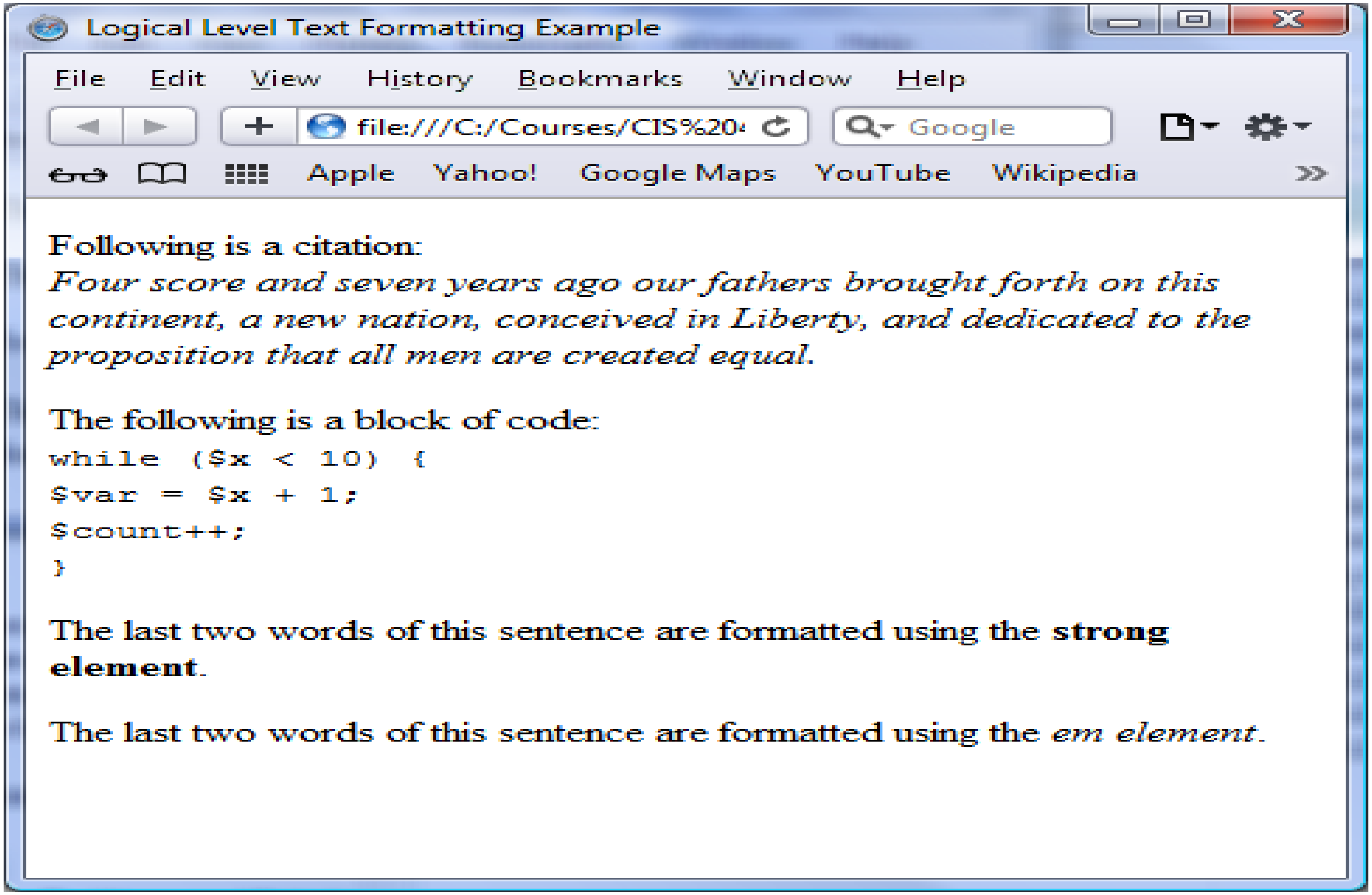

linebreak.html
heading.html
horizontal rule.html
div.html
presentation level text formatting.html
logical level text formatting.html

```
<!DOCTYPE html>
<html>
   <head>
     <title>Logical Level Text Formatting Example</title>
   </head>
   <body>
    <p>Following is a citation: <br />
       <cite>
           Four score and seven years ago our fathers brought
           forth on this continent, a new nation, conceived in Liberty,
           and dedicated to the proposition that all men are created equal.
       </cite>
    </p>
    <p>The following is a block of code: <br />
       <code>
           while ($x &lt; 10) { <br />
              $var = $x + 1; <br />
              $count++; <br />
             } <br />
        </code>
    </p>
    <p>The last two words of this sentence are formatted using the <strong> strong
       element</strong>.
    </p>
    <p>The last two words of this sentence are formatted using the
        <em> em element</em>.</p>
   </body>
</html>
```

Hyper Text Markup Language file   length : 850   lines : 30   Ln : 2   Col : 6   Sel : 0   Dos\Windows   ANSI   INS

See page 71 for more details

Presentational Text Formatting – Example


Logical Level Text Formatting Example
File Edit View History Bookmarks Window Help
file:///C:/Courses/CIS%20
Google
Apple Yahoo! Google Maps YouTube Wikipedia

Following is a citation:
Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and dedicated to the proposition that all men are created equal.

The following is a block of code:

```
while ($x < 10) {
$var = $x + 1;
$count++;
}
```

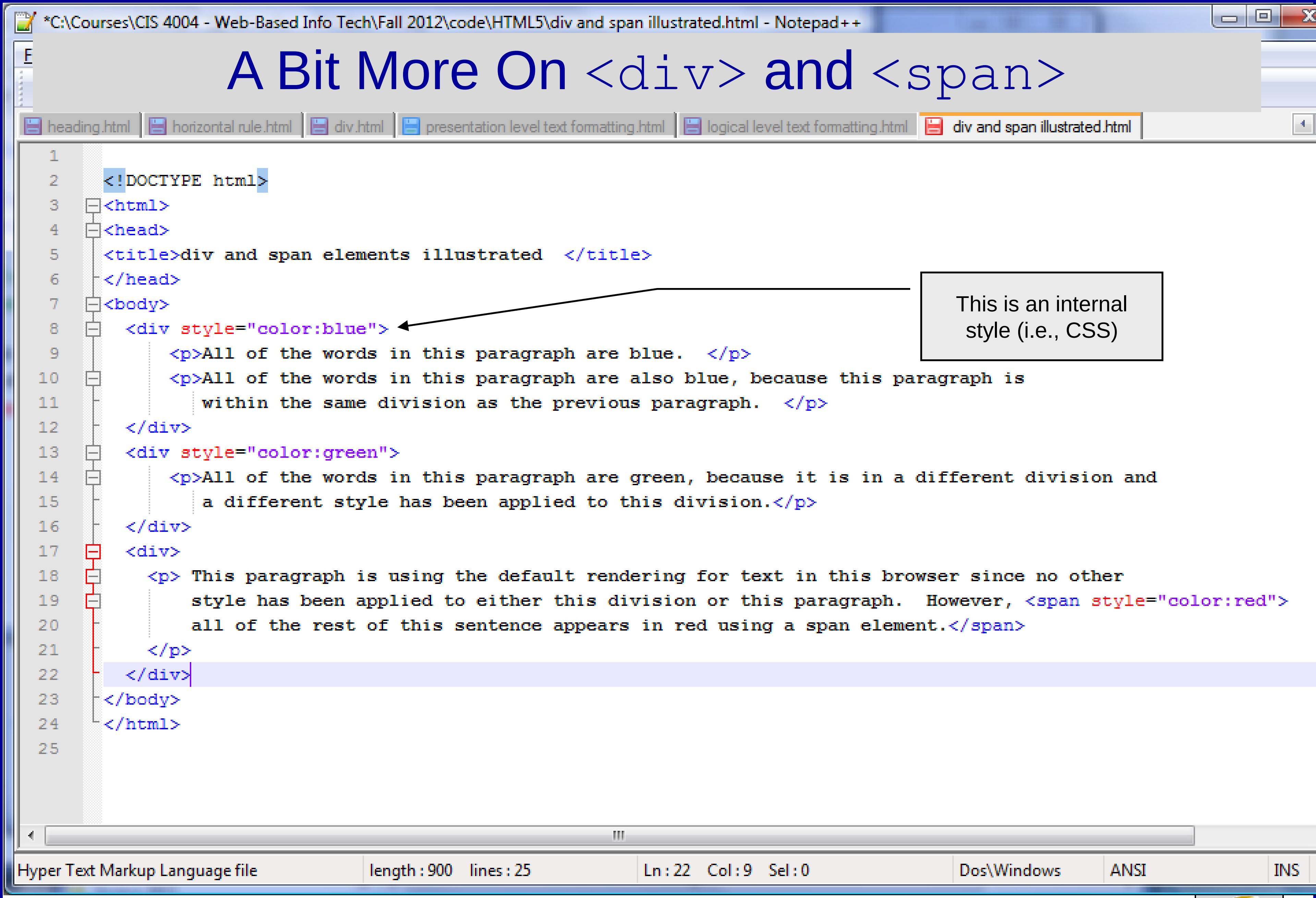
The last two words of this sentence are formatted using the **strong element**.

The last two words of this sentence are formatted using the *em element*.

A Bit More On `<div>` and `<span>`

- The `<div>` and `<span>` elements are most often used in conjunction with style sheets (as we will see later) to avoid the default rendering of elements.
- `<div>` is a block element whereas `<span>` is an inline element. Neither of these elements have a default rendering, which in a sense makes them generic tags. Since they have no default rendering, they are very useful for arbitrary style duties.
- A `<div>` element induces a hard return while the inline `<span>` element does not.
- The following example will more clearly illustrate the differences between these two elements.

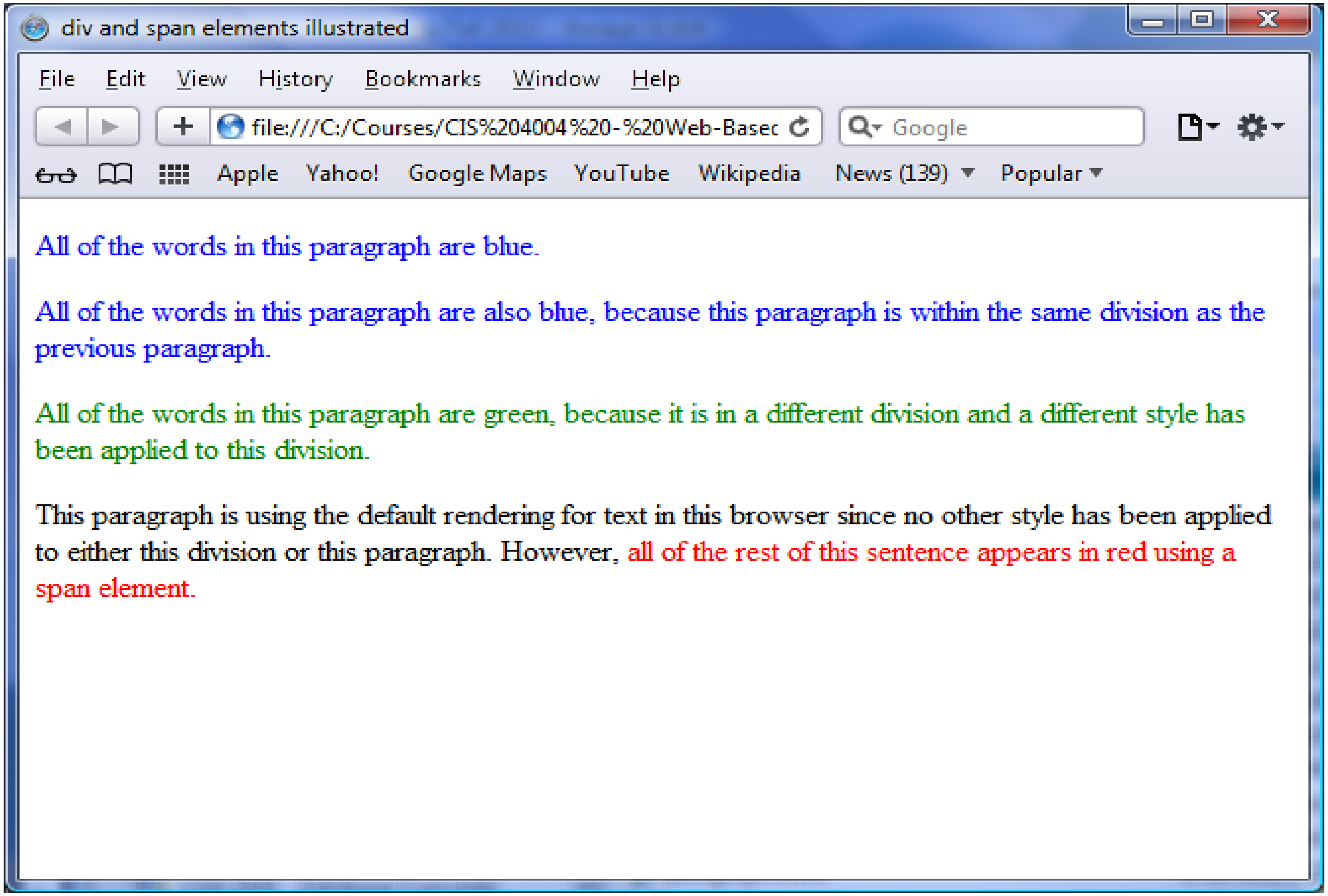
A Bit More On `<div>` and `<span>`


*C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\div and span illustrated.html - Notepad++
heading.html
horizontal rule.html
div.html
presentation level text formatting.html
logical level text formatting.html
div and span illustrated.html
Hyper Text Markup Language file
length : 900   lines : 25
Ln : 22   Col : 9   Sel : 0
Dos\Windows
ANSI
INS

```html

<!DOCTYPE html>
<html>
<head>
<title>div and span elements illustrated  </title>
</head>
<body>
  <div style="color:blue">
      <p>All of the words in this paragraph are blue.  </p>
      <p>All of the words in this paragraph are also blue, because this paragraph is
         within the same division as the previous paragraph.  </p>
  </div>
  <div style="color:green">
      <p>All of the words in this paragraph are green, because it is in a different division and
         a different style has been applied to this division.</p>
  </div>
  <div>
    <p> This paragraph is using the default rendering for text in this browser since no other
        style has been applied to either this division or this paragraph.  However, <span style="color:red">
        all of the rest of this sentence appears in red using a span element.</span>
    </p>
  </div>
</body>
</html>

```

This is an internal style (i.e., CSS)

A Bit More On `<div>` and `<span>`



div and span elements illustrated
File Edit View History Bookmarks Window Help
file:///C:/Courses/CIS%204004%20-%20Web-Basec
Google
Apple Yahoo! Google Maps YouTube Wikipedia News (139) Popular

All of the words in this paragraph are blue.

All of the words in this paragraph are also blue, because this paragraph is within the same division as the previous paragraph.

All of the words in this paragraph are green, because it is in a different division and a different style has been applied to this division.

This paragraph is using the default rendering for text in this browser since no other style has been applied to either this division or this paragraph. However, all of the rest of this sentence appears in red using a span element.

Character Entities In XHTML

- The character of the less than symbol shown in the example on page 29 is written as `&lt;`.
- Certain characters in XHTML have special meaning to parser, like less than < and greater than >. These characters identify the beginning and ending of a tag, so if you want to add these characters as literal values, you must use the character entity code for them.
- A character entity is written in the following syntax: `&code;`. It begins with an ampersand (&) character, then the code for the entity, then a semicolon (;).
- Hundreds of symbols can be referenced and included on Web pages using entities. Some of the more popular symbols can be referenced by their abbreviations, like less than (`lt`) and greater than (`gt`), but they can also be referenced using their decimal value in the ASCII Table.
- Some of the most common character entity codes are shown in the table on the next page.

Some Common Character Entity Codes

Symbol	Description	XHTML Code
>	Greater than	`&gt;   or   &62;`
<	Less than	`&lt;   or   &60;`
®	Trademark	`&trade;   or   &174;`
©	Copyright	`&copy;   or   &169;`
¢	Cent sign	`&cent;   or   &162;`
	Non-breaking space	`&nbsp;   or   &160`

HTML5 Lists

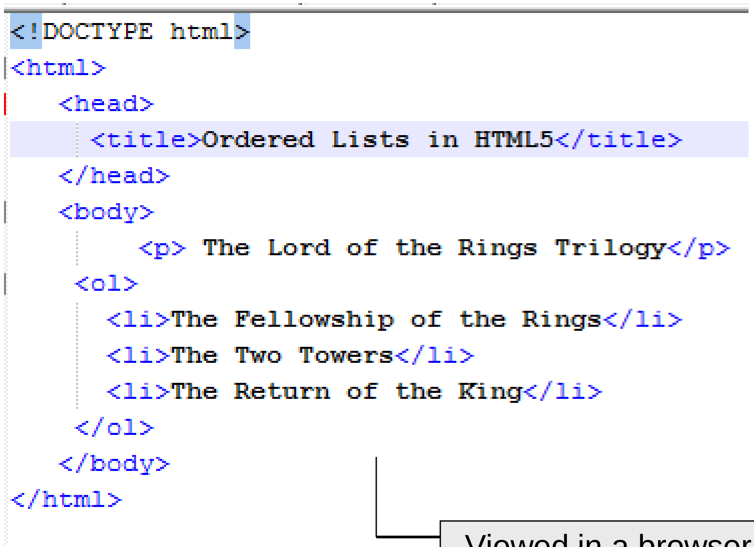
- HTML5 provides three main types of lists: numbered, bulleted, and definition. These are summarized in the table below.

List type	XHTML element	Item element
Ordered list (numbered)	`<ol>…</ol>`	`<li>...</li>`
Unordered list (bulleted)	`<ul>…</ul>`	`<li>...</li>`

Definition list	`<dl>…</dl>`	`<dt>...</dt>`
		and
		`<dd>...</dd>`

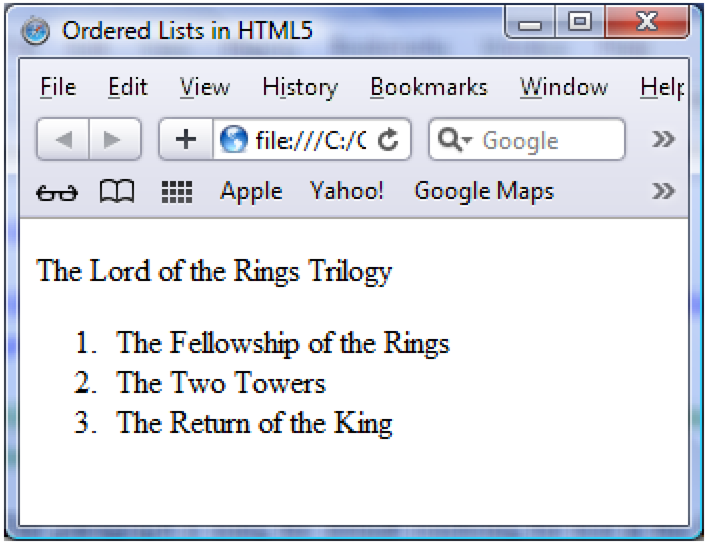
Ordered Lists

- Ordered lists are numbered and are contained within the `<ol>…</ol>` , ordered list element.
- An ordered list may have any number of `<li>` list items appearing in the element content.
- A browser will list each of the elements in a number sequential list.
- For example:



```
<!DOCTYPE html>
<html>
   <head>
     <title>Ordered Lists in HTML5</title>
   </head>
   <body>
        <p> The Lord of the Rings Trilogy</p>
    <ol>
      <li>The Fellowship of the Rings</li>
      <li>The Two Towers</li>
      <li>The Return of the King</li>
    </ol>
   </body>
</html>
```

Viewed in a browser

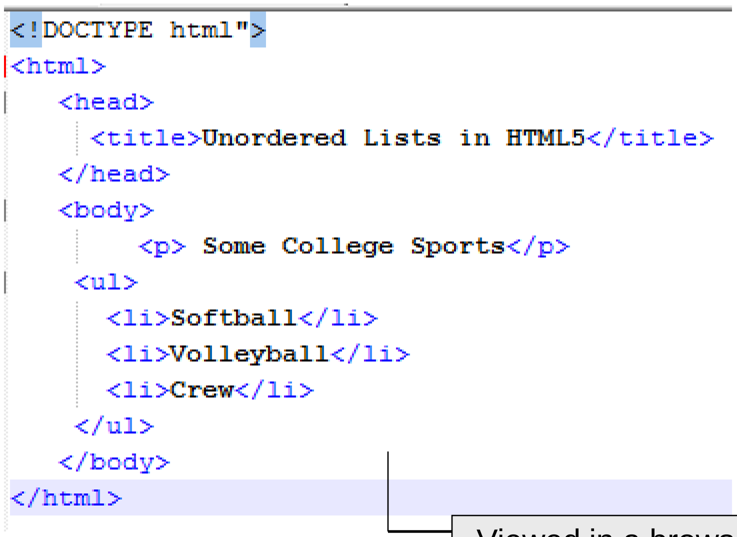


The Lord of the Rings Trilogy

1. The Fellowship of the Rings
2. The Two Towers
3. The Return of the King

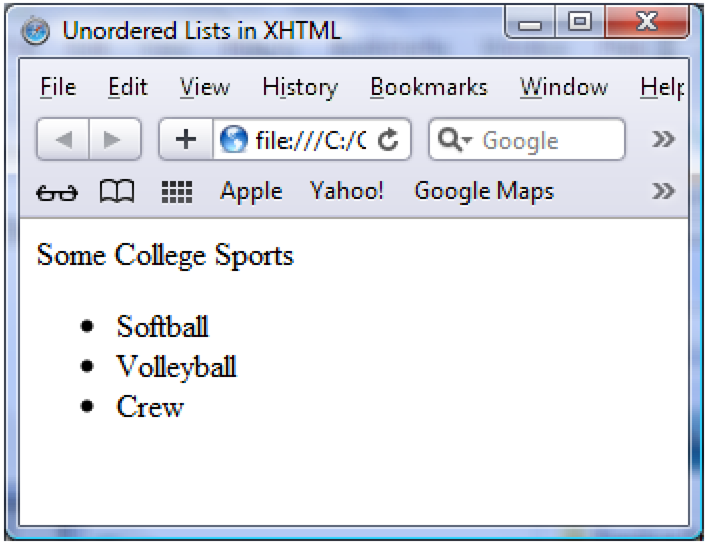
Unordered Lists

- Unordered lists are bulleted instead of numbered. An unordered list is contained within the `<ul>…</ul>` , unordered list element.
- An unordered list may have any number of `<li>` list items appearing in the element content.
- A browser will list each of the elements in a bulleted list.
- For example:



```
<!DOCTYPE html">
<html>
   <head>
     <title>Unordered Lists in HTML5</title>
   </head>
   <body>
        <p> Some College Sports</p>
    <ul>
      <li>Softball</li>
      <li>Volleyball</li>
      <li>Crew</li>
    </ul>
   </body>
</html>
```

Viewed in a browser



Some College Sports

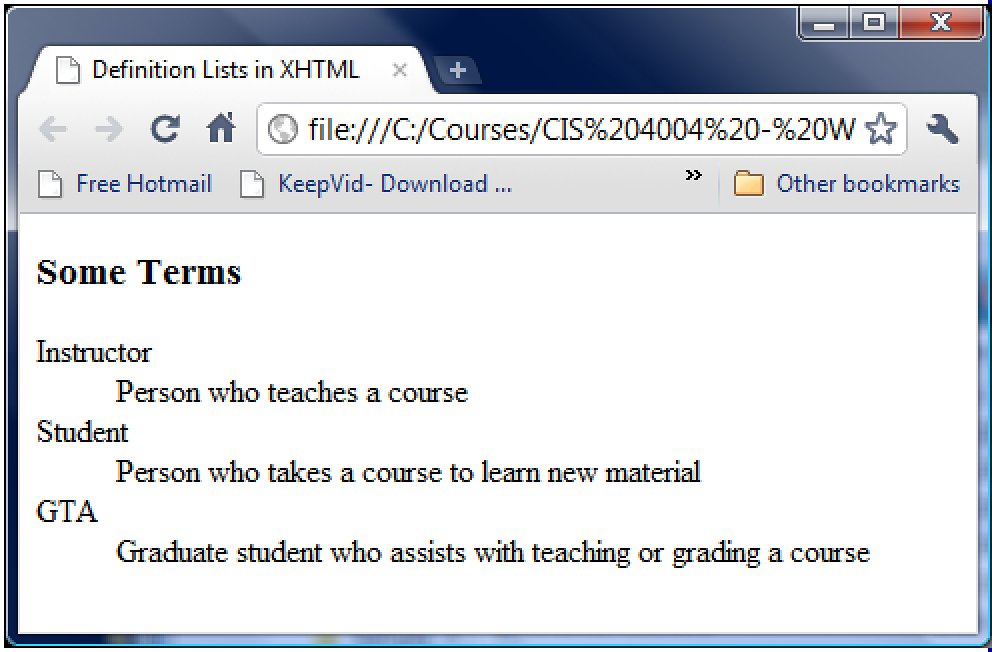
- Softball
- Volleyball
- Crew

Definition Lists

- Definition lists are lists of terms and their definitions. They are a little different than ordered and unordered lists in that the items are lists in pairs.
- The `<dl>…</dl>` , surround the definition list. The name of the term appears between `<dt>` and `</dt>` tags, and the definition is between `<dd>` and `</dd>` tags.
- For example:

```
<h3> Some Terms</h3>
<dl>
  <dt>Instructor</dt><dd>Person who
      teaches a course</dd>
  <dt>Student</dt><dd>Person who
      takes a course to learn new
      material</dd>
  <dt>GTA</dt><dd>Graduate student
      who assists with teaching a
      course</dd>
</dl>
```

Viewed in a browser


Definition Lists in XHTML
file:///C:/Courses/CIS%204004%20-%20W
Free Hotmail
KeepVid- Download ...
Other bookmarks

Some Terms

Instructor

Person who teaches a course

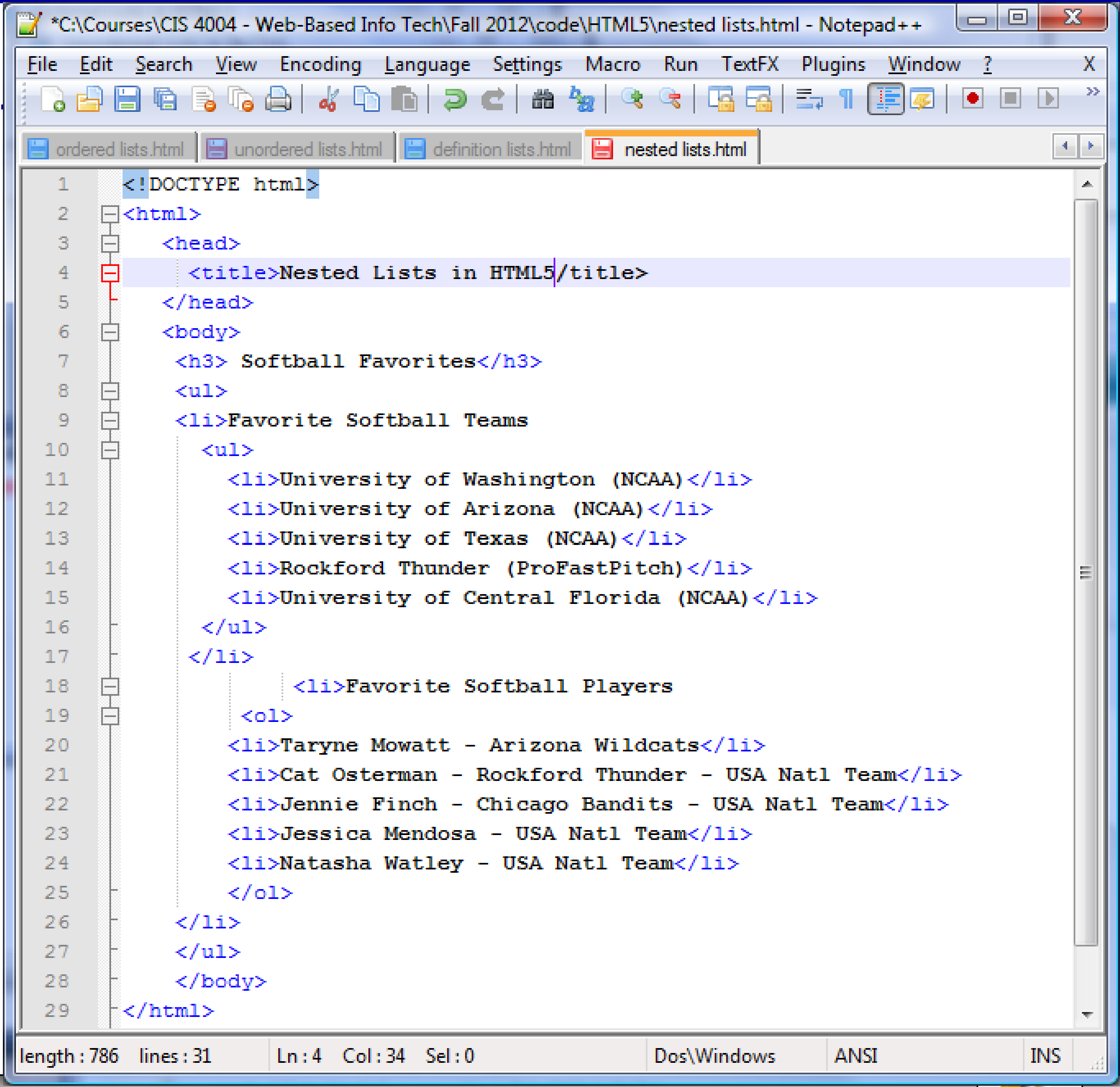
Student

Person who takes a course to learn new material

GTA

Graduate student who assists with teaching or grading a course

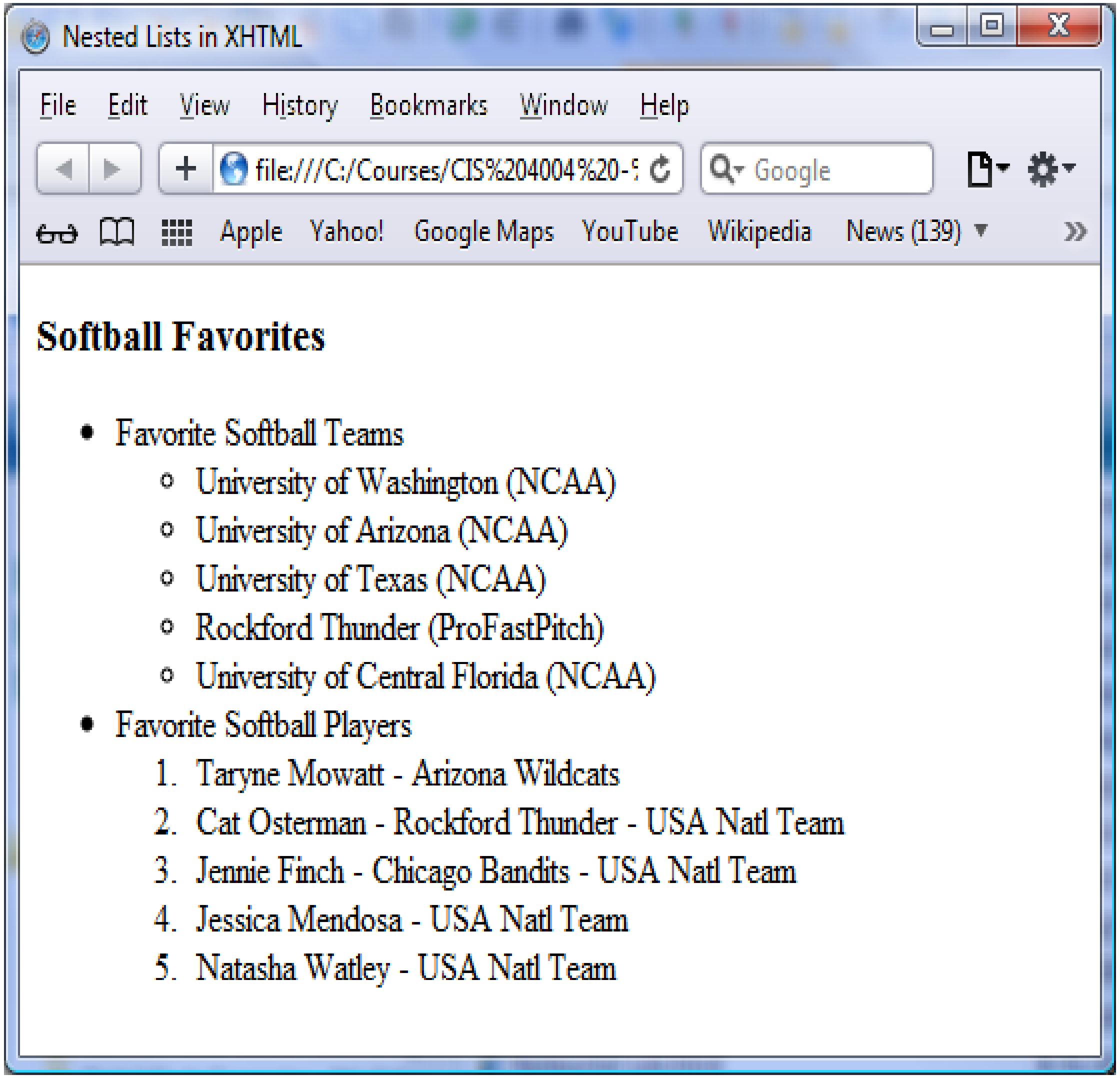
Nesting Lists Example


*C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\nested lists.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
ordered lists.html
unordered lists.html
definition lists.html
nested lists.html

```
<!DOCTYPE html>
<html>
   <head>
     <title>Nested Lists in HTML5</title>
   </head>
   <body>
    <h3> Softball Favorites</h3>
    <ul>
    <li>Favorite Softball Teams
      <ul>
        <li>University of Washington (NCAA)</li>
        <li>University of Arizona (NCAA)</li>
        <li>University of Texas (NCAA)</li>
        <li>Rockford Thunder (ProFastPitch)</li>
        <li>University of Central Florida (NCAA)</li>
      </ul>
     </li>
             <li>Favorite Softball Players
         <ol>
        <li>Taryne Mowatt - Arizona Wildcats</li>
        <li>Cat Osterman - Rockford Thunder - USA Natl Team</li>
        <li>Jennie Finch - Chicago Bandits - USA Natl Team</li>
        <li>Jessica Mendosa - USA Natl Team</li>
        <li>Natasha Watley - USA Natl Team</li>
        </ol>
    </li>
    </ul>
    </body>
</html>
```

length : 786 lines : 31 Ln : 4 Col : 34 Sel : 0 Dos\Windows ANSI INS

Nesting Lists – Example


Nested Lists in XHTML
File Edit View History Bookmarks Window Help
file:///C:/Courses/CIS%204004%20-
Google
Apple Yahoo! Google Maps YouTube Wikipedia News (139)

Softball Favorites

- Favorite Softball Teams
  - University of Washington (NCAA)
  - University of Arizona (NCAA)
  - University of Texas (NCAA)
  - Rockford Thunder (ProFastPitch)
  - University of Central Florida (NCAA)
- Favorite Softball Players
  1. Taryne Mowatt - Arizona Wildcats
  2. Cat Osterman - Rockford Thunder - USA Natl Team
  3. Jennie Finch - Chicago Bandits - USA Natl Team
  4. Jessica Mendosa - USA Natl Team
  5. Natasha Watley - USA Natl Team

Hyperlinks

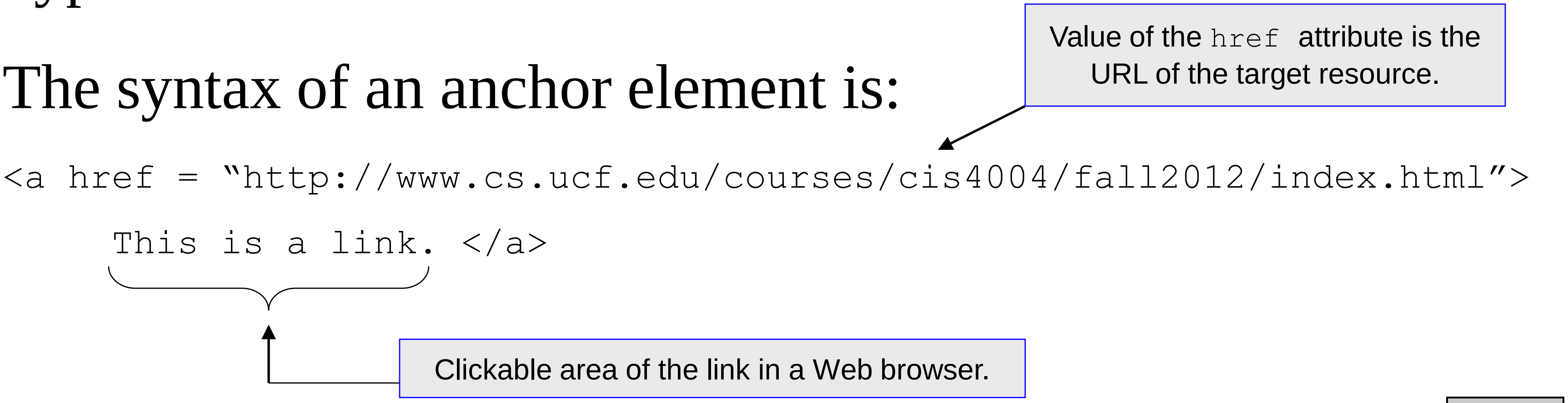
- A hypertext link, or hyperlink is an object in a Web page that when clicked will redirect the browser to another Web page or file.
- Usually, hyperlinks take the form of blue, underlined text, or an image.
- Special linking elements are included in the XHTML (also in HTML) specification that allow Web page authors to use images or text within a We page to create these links to other resources. The resource being linked to by the hyperlink is called the target resource.
- In addition to other Web pages, the target resource can be an image file, a multimedia file (such as an audio or video file), another section within the same page, or any Web page or file anywhere on the Internet.
- Hyperlinks provide Web page authors with a powerful means of organizing information and allow them to create very complex, cross-referenced Web sites with clickable tables of contents and menus.

Creating Hyperlinks With The `<a>` Element

- The `<a>` or anchor element in XHTML is used to create hyperlinks. These links require the user to perform an action – usually clicking on the link – in order to for the link to do anything. The clickable region of the link can consist of text or images.

- If the user never clicks the linked image or text, the link is never activated. Passively moving the cursor over the hyperlink will not activate it.

- The syntax of an anchor element is:

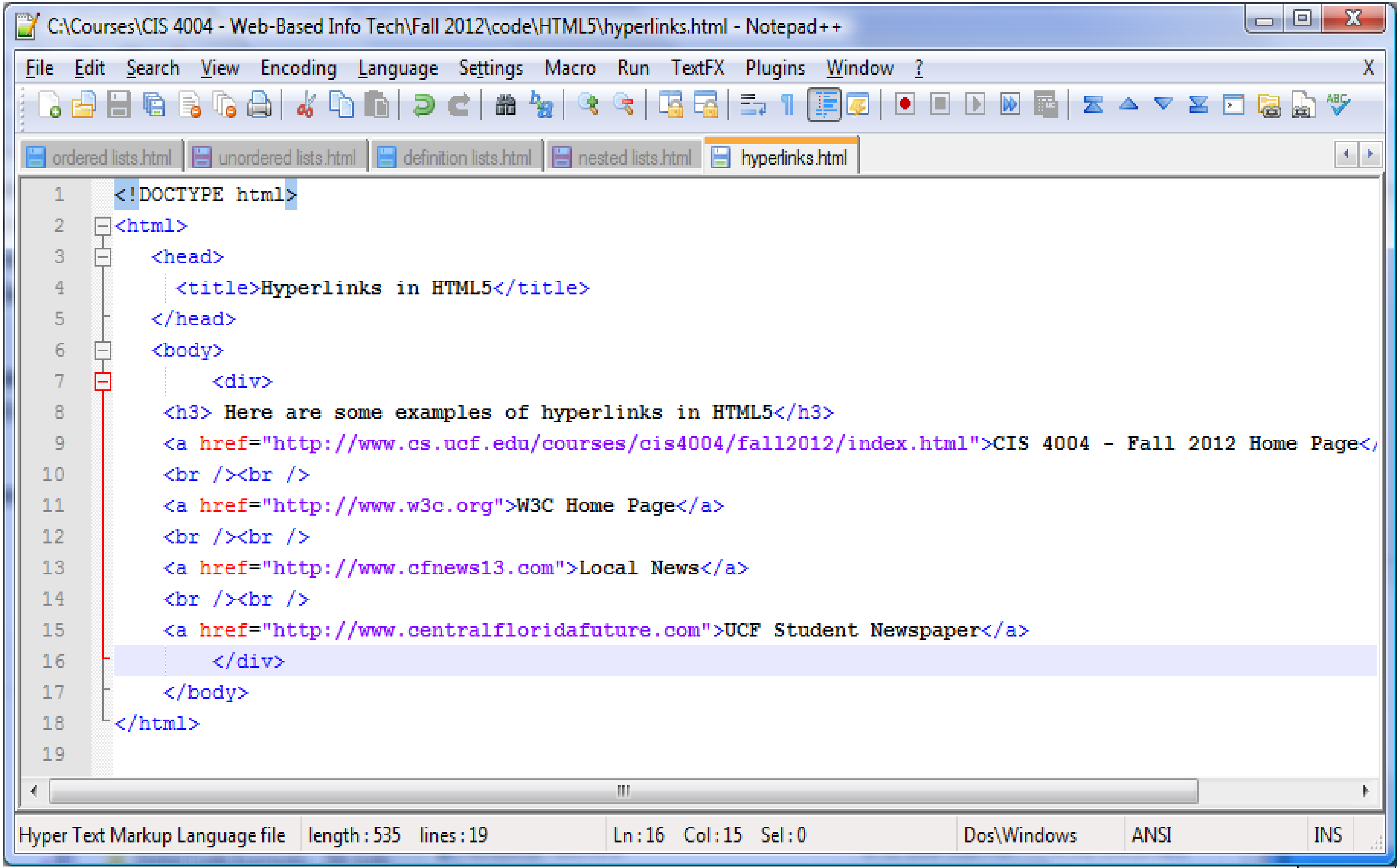


Value of the `href` attribute is the URL of the target resource.

```
<a href = “http://www.cs.ucf.edu/courses/cis4004/fall2012/index.html”>
      This is a link. </a>
```

Clickable area of the link in a Web browser.

Hyperlink – Example

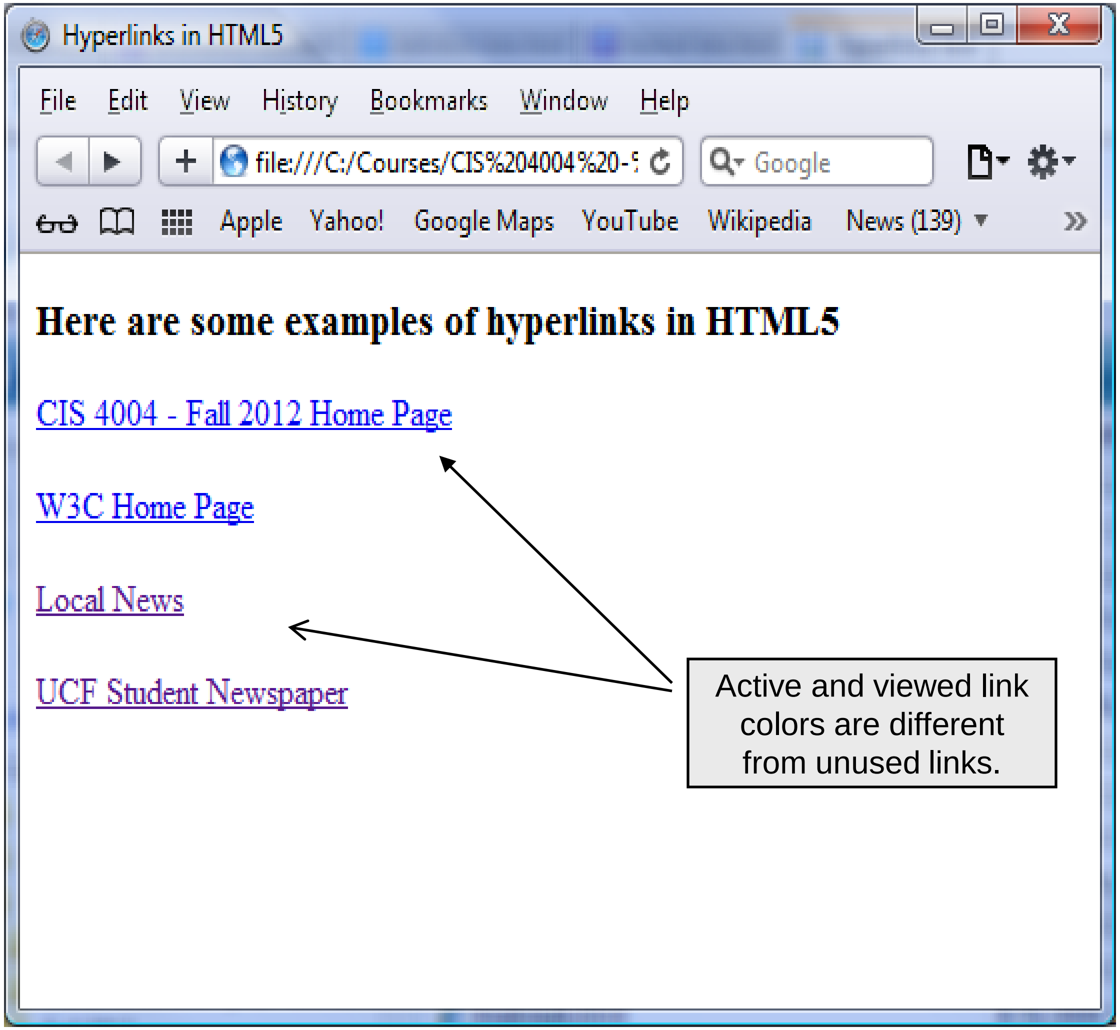


C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\hyperlinks.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
ordered lists.html
unordered lists.html
definition lists.html
nested lists.html
hyperlinks.html

```
<!DOCTYPE html>
<html>
   <head>
     <title>Hyperlinks in HTML5</title>
   </head>
   <body>
        <div>
    <h3> Here are some examples of hyperlinks in HTML5</h3>
    <a href="http://www.cs.ucf.edu/courses/cis4004/fall2012/index.html">CIS 4004 - Fall 2012 Home Page</
    <br /><br />
    <a href="http://www.w3c.org">W3C Home Page</a>
    <br /><br />
    <a href="http://www.cfnews13.com">Local News</a>
    <br /><br />
    <a href="http://www.centralfloridafuture.com">UCF Student Newspaper</a>
        </div>
    </body>
</html>
```

Hyper Text Markup Language file
length : 535   lines : 19
Ln : 16   Col : 15   Sel : 0
Dos\Windows
ANSI
INS

Hyperlink – Example



Hyperlinks in HTML5
File Edit View History Bookmarks Window Help
file:///C:/Courses/CIS%204004%20-
Google
Apple Yahoo! Google Maps YouTube Wikipedia News (139)

Here are some examples of hyperlinks in HTML5

CIS 4004 - Fall 2012 Home Page

W3C Home Page

Local News

UCF Student Newspaper

Active and viewed link colors are different from unused links.

Relative versus Absolute URLs

- Relative URLs are used to link documents that reside on the same Web server. When a relative URL is used, the protocol and domain name are omitted. The link to the target resource is relative to the location of the document containing the link, or the source document.
- If the target resource resides in the same directory as the source document, you can use a link containing only the name of the target resource, as in the first example below.
- If the target resource resides in a different directory on the Web server, you must include the subdirectory information in the link, as in the second example below.

```
<a href=“newpage.html”>Click Here</a>

<a href=“documents/newpage.html”>Click Here</a>

<a href=“images/mypicture.jpg”>Click Here</a>
```

Relative versus Absolute URLs

- Absolute URLs are used to link documents that reside on different Web servers. When an absolute URL is used, the protocol (`http://`) and domain name (`cs.ucf.edu`) and domain name are included to direct the Web browser to the location of the new Web server. The absolute URL does not take into account any location information about the current document and can reference any target resource anywhere on the Internet.

- Below are some examples.

```
<a href="http://www.cs.ucf.edu/courses/cis
4004/fall2012/index.html>">Click Here</a>

<a
href="http://www.cs.ucf.edu/courses/cis4004/fall2012/background.gi
f"> Click Here</a>
```

Linking Within A Single Document

- If you are working with a large document, you may want to create links to sections within that document.

- For example, you may want to create a link at the bottom of the document that links to back to the top of the document, or a link that will take you to a footnote at the bottom of a page from within the body of the document.

- You see internal linking quite often when viewing on-line tutorials, or documentation in which each chapter is linked from one to the next and even pages within a chapter are linked from one to the next.

Linking Within A Single Document

- In order to create an internal link, you will need to first create the anchor at the place where you want the link to link to. The anchor element is used with an attribute called name, which identifies the anchor, or target.

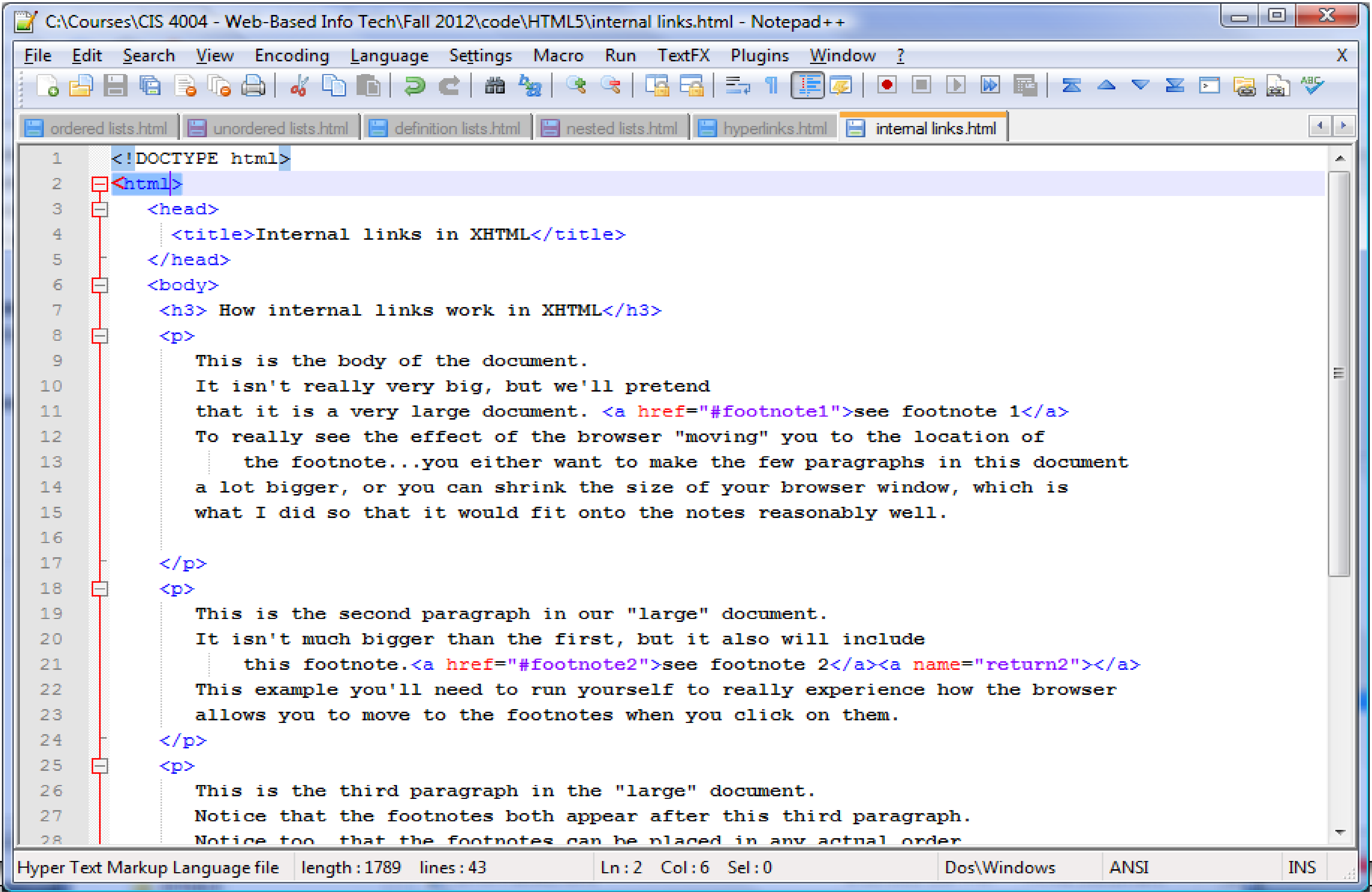
```
<a name="footnote">Footnote</a>
```

- Next, you need to create a link that looks like the relative links we've already examined, but has a # sign in front of the relative URL to tell the browser that this link exists in the current document. This link would look like:

```
<a href="#footnote">Link to footnote</a>.
```

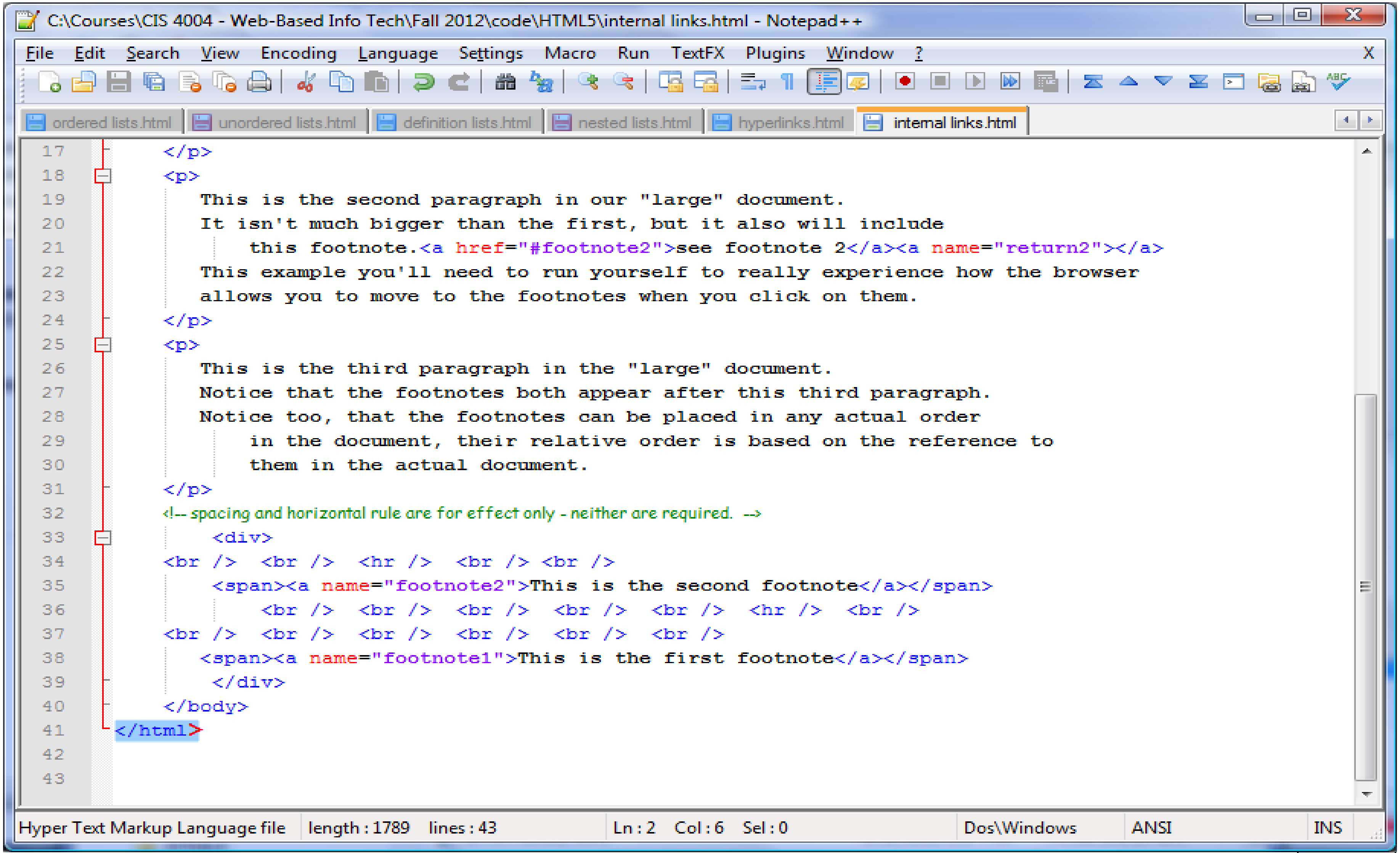
- This would create an anchor where the footnote resides in the document, and clicking on the link would then take the user to that place within the document. The example on the next couple of pages illustrates internal links.

Linking Within A Single Document


C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\internal links.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
ordered lists.html
unordered lists.html
definition lists.html
nested lists.html
hyperlinks.html
internal links.html
Hyper Text Markup Language file
length : 1789 lines : 43
Ln : 2 Col : 6 Sel : 0
Dos\Windows
ANSI
INS

```
<!DOCTYPE html>
<html>
   <head>
     <title>Internal links in XHTML</title>
   </head>
   <body>
    <h3> How internal links work in XHTML</h3>
    <p>
       This is the body of the document.
       It isn't really very big, but we'll pretend
       that it is a very large document. <a href="#footnote1">see footnote 1</a>
       To really see the effect of the browser "moving" you to the location of
           the footnote...you either want to make the few paragraphs in this document
       a lot bigger, or you can shrink the size of your browser window, which is
       what I did so that it would fit onto the notes reasonably well.

    </p>
    <p>
       This is the second paragraph in our "large" document.
       It isn't much bigger than the first, but it also will include
           this footnote.<a href="#footnote2">see footnote 2</a><a name="return2"></a>
       This example you'll need to run yourself to really experience how the browser
       allows you to move to the footnotes when you click on them.
    </p>
    <p>
       This is the third paragraph in the "large" document.
       Notice that the footnotes both appear after this third paragraph.
       Notice too, that the footnotes can be placed in any actual order
```

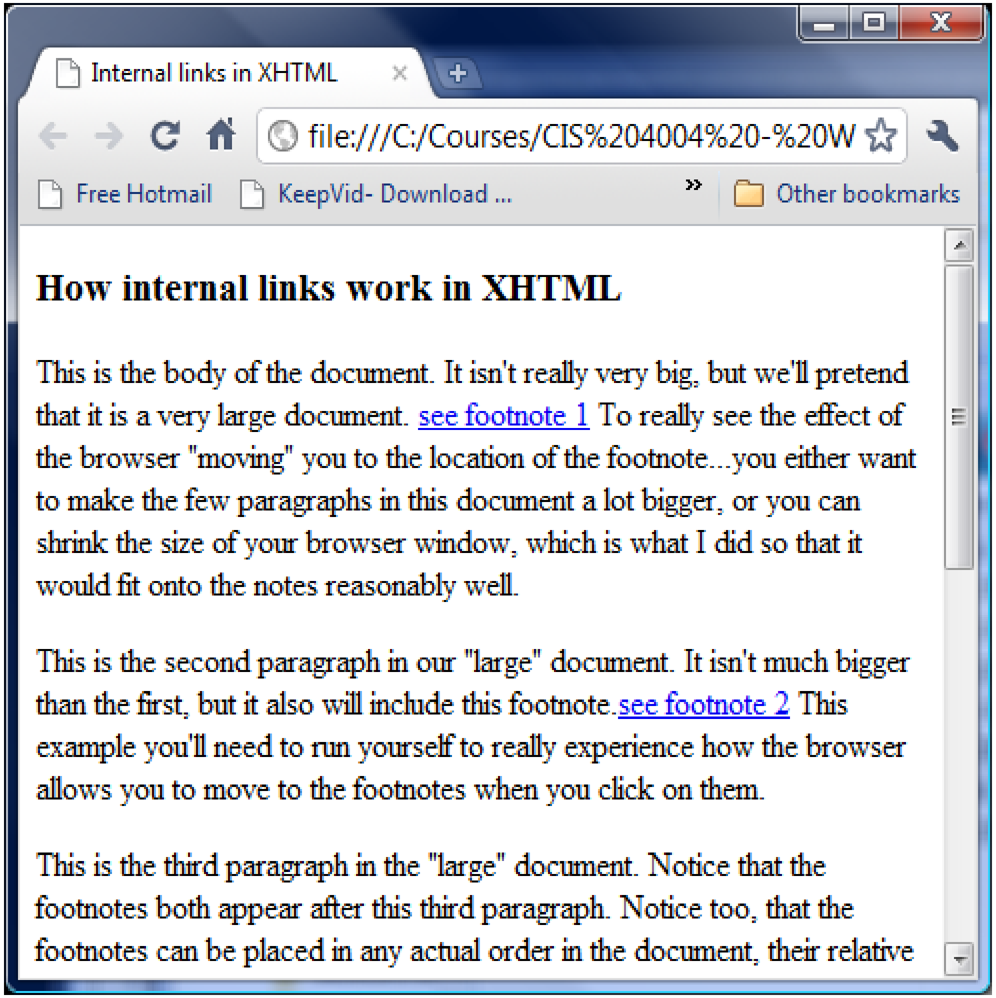
Linking Within A Single Document



```
    </p>
    <p>
       This is the second paragraph in our "large" document.
       It isn't much bigger than the first, but it also will include
           this footnote.<a href="#footnote2">see footnote 2</a><a name="return2"></a>
       This example you'll need to run yourself to really experience how the browser
       allows you to move to the footnotes when you click on them.
    </p>
    <p>
       This is the third paragraph in the "large" document.
       Notice that the footnotes both appear after this third paragraph.
       Notice too, that the footnotes can be placed in any actual order
           in the document, their relative order is based on the reference to
           them in the actual document.
    </p>
    <!-- spacing and horizontal rule are for effect only - neither are required.  -->
        <div>
    <br />  <br />  <hr />  <br /> <br />
        <span><a name="footnote2">This is the second footnote</a></span>
            <br />  <br />  <br />  <br />  <br />  <hr />  <br />
    <br />  <br />  <br />  <br />  <br />  <br />
       <span><a name="footnote1">This is the first footnote</a></span>
        </div>
    </body>
</html>
```

Hyper Text Markup Language file   length : 1789   lines : 43   Ln : 2   Col : 6   Sel : 0   Dos\Windows   ANSI   INS

Linking Within A Single Document



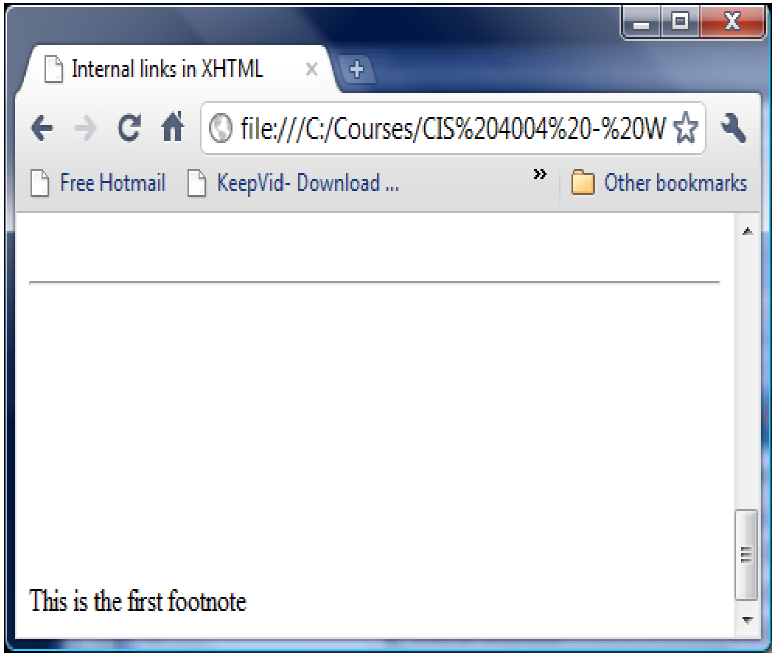
How internal links work in XHTML

This is the body of the document. It isn't really very big, but we'll pretend that it is a very large document. see footnote 1 To really see the effect of the browser "moving" you to the location of the footnote...you either want to make the few paragraphs in this document a lot bigger, or you can shrink the size of your browser window, which is what I did so that it would fit onto the notes reasonably well.

This is the second paragraph in our "large" document. It isn't much bigger than the first, but it also will include this footnote.see footnote 2 This example you'll need to run yourself to really experience how the browser allows you to move to the footnotes when you click on them.

This is the third paragraph in the "large" document. Notice that the footnotes both appear after this third paragraph. Notice too, that the footnotes can be placed in any actual order in the document, their relative

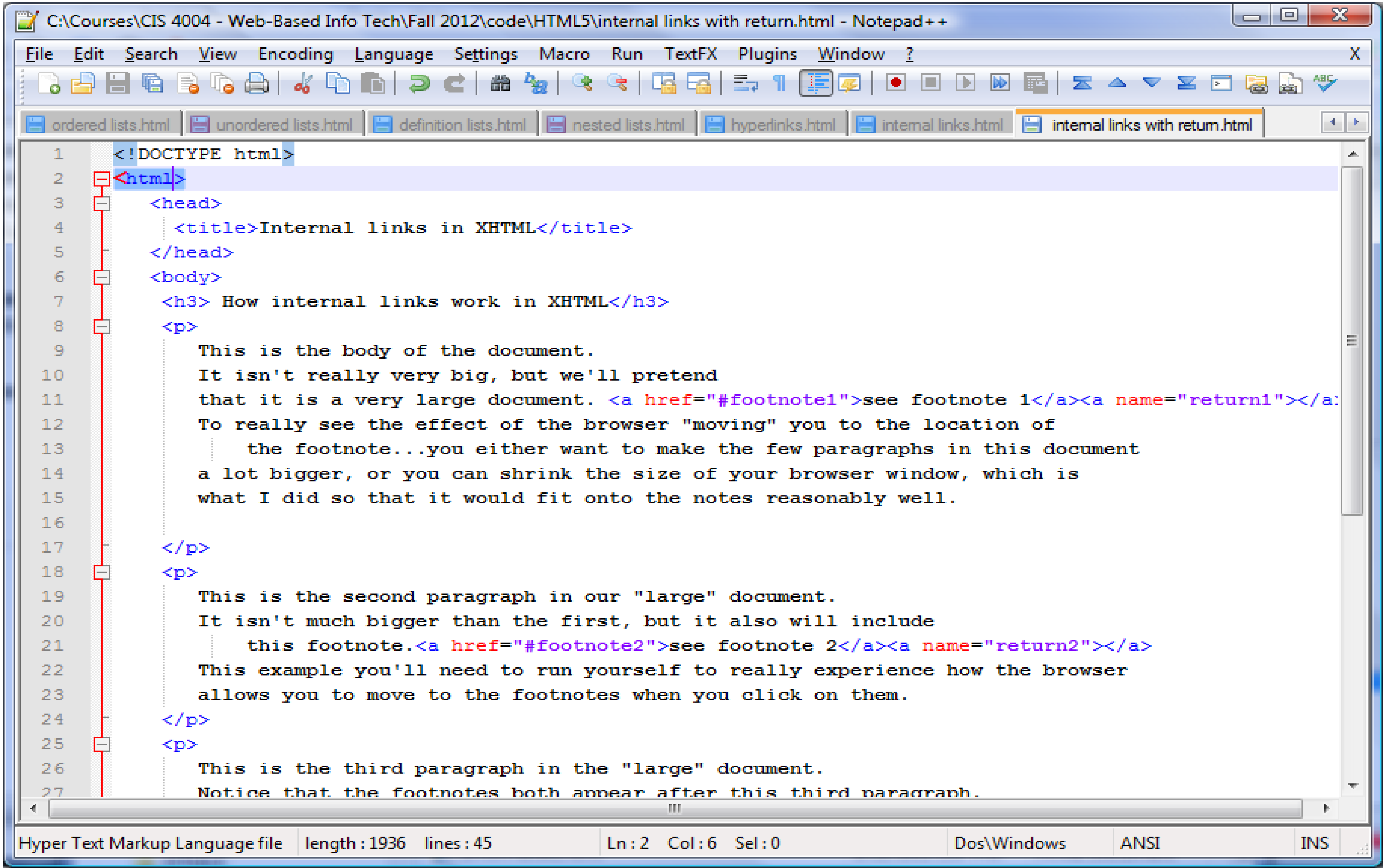
Initial browser window



This is the first footnote

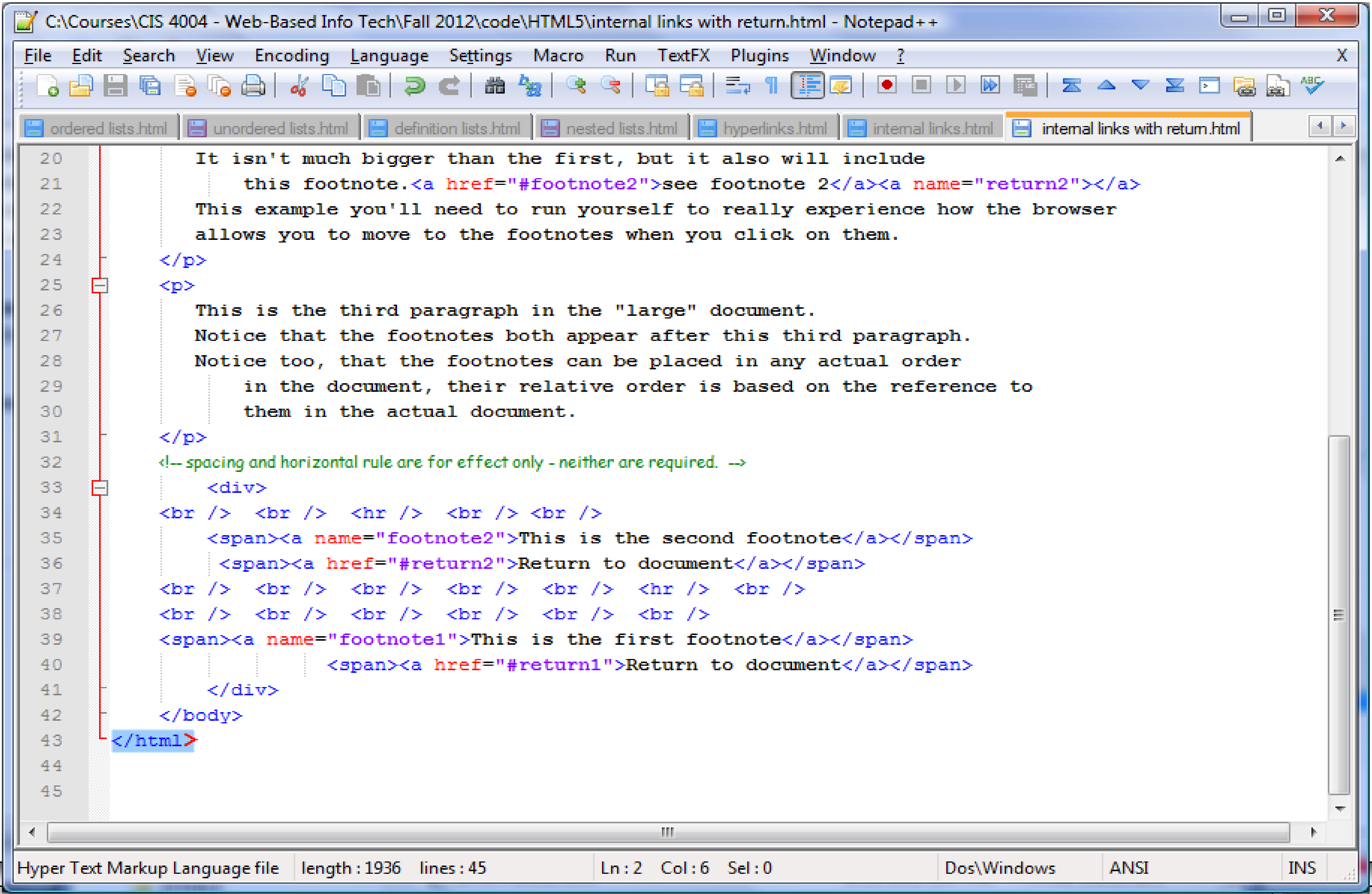
After clicking footnote #1 link

Linking Within A Single Document - Returning



```html
<!DOCTYPE html>
<html>
   <head>
     <title>Internal links in XHTML</title>
   </head>
   <body>
    <h3> How internal links work in XHTML</h3>
    <p>
       This is the body of the document.
       It isn't really very big, but we'll pretend
       that it is a very large document. <a href="#footnote1">see footnote 1</a><a name="return1"></a
       To really see the effect of the browser "moving" you to the location of
           the footnote...you either want to make the few paragraphs in this document
       a lot bigger, or you can shrink the size of your browser window, which is
       what I did so that it would fit onto the notes reasonably well.

    </p>
    <p>
       This is the second paragraph in our "large" document.
       It isn't much bigger than the first, but it also will include
           this footnote.<a href="#footnote2">see footnote 2</a><a name="return2"></a>
       This example you'll need to run yourself to really experience how the browser
       allows you to move to the footnotes when you click on them.
    </p>
    <p>
       This is the third paragraph in the "large" document.
       Notice that the footnotes both appear after this third paragraph.
```

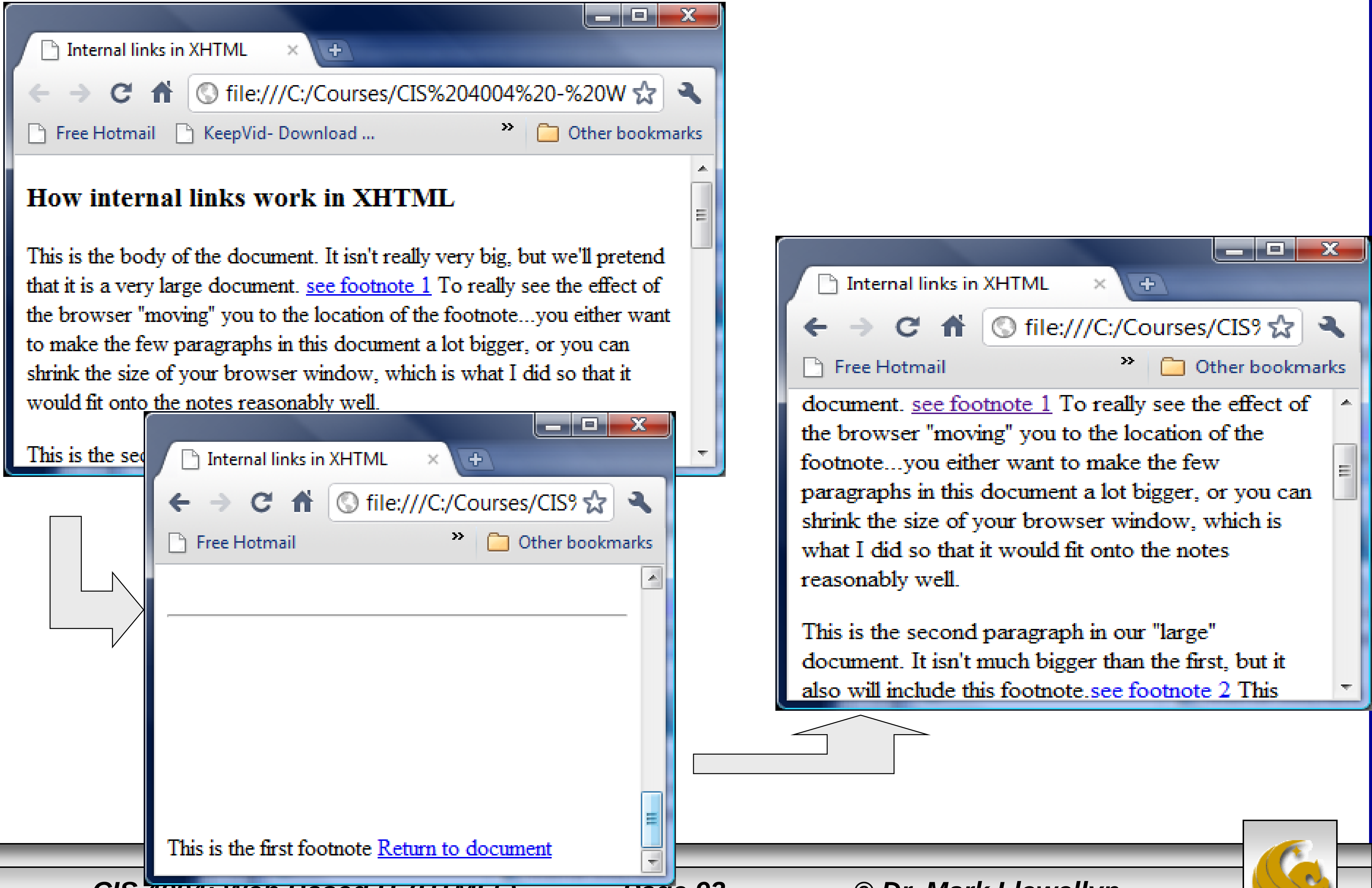
Linking Within A Single Document - Returning


C:\Courses\CIS 4004 - Web-Based Info Tech\Fall 2012\code\HTML5\internal links with return.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
ordered lists.html
unordered lists.html
definition lists.html
nested lists.html
hyperlinks.html
internal links.html
internal links with return.html

```
   It isn't much bigger than the first, but it also will include
      this footnote.<a href="#footnote2">see footnote 2</a><a name="return2"></a>
   This example you'll need to run yourself to really experience how the browser
   allows you to move to the footnotes when you click on them.
  </p>
  <p>
   This is the third paragraph in the "large" document.
   Notice that the footnotes both appear after this third paragraph.
   Notice too, that the footnotes can be placed in any actual order
      in the document, their relative order is based on the reference to
      them in the actual document.
  </p>
  <!-- spacing and horizontal rule are for effect only - neither are required.  -->
    <div>
  <br />  <br />  <hr />  <br /> <br />
    <span><a name="footnote2">This is the second footnote</a></span>
     <span><a href="#return2">Return to document</a></span>
  <br />  <br />  <br />  <br />  <br />  <hr />  <br />
  <br />  <br />  <br />  <br />  <br />  <br />
  <span><a name="footnote1">This is the first footnote</a></span>
         <span><a href="#return1">Return to document</a></span>
    </div>
  </body>
</html>
```

Hyper Text Markup Language file   length : 1936   lines : 45   Ln : 2   Col : 6   Sel : 0   Dos\Windows   ANSI   INS

Linking Within A Single Document - Returning



How internal links work in XHTML

This is the body of the document. It isn't really very big, but we'll pretend that it is a very large document. see footnote 1 To really see the effect of the browser "moving" you to the location of the footnote...you either want to make the few paragraphs in this document a lot bigger, or you can shrink the size of your browser window, which is what I did so that it would fit onto the notes reasonably well.

This is the se

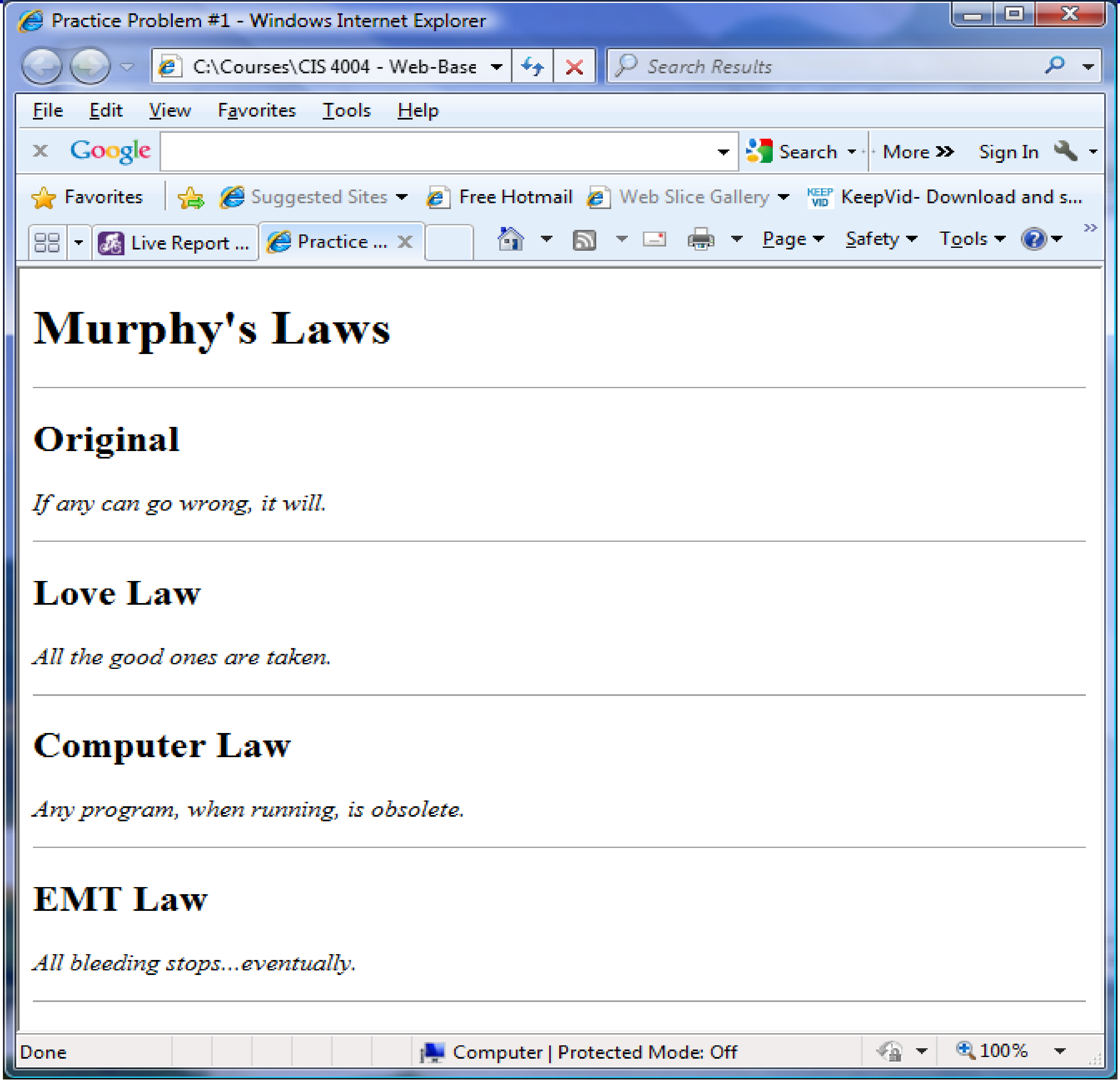
This is the first footnote Return to document

document. see footnote 1 To really see the effect of the browser "moving" you to the location of the footnote...you either want to make the few paragraphs in this document a lot bigger, or you can shrink the size of your browser window, which is what I did so that it would fit onto the notes reasonably well.

This is the second paragraph in our "large" document. It isn't much bigger than the first, but it also will include this footnote.see footnote 2 This

A Practice Problem

- Use the text file named “murphy.txt” available on the course website and mark it up so that when rendered in a browser it looks like the following page.


Practice Problem #1 - Windows Internet Explorer
C:\Courses\CIS 4004 - Web-Base
Search Results
File Edit View Favorites Tools Help
Google
Search
More »
Sign In
Favorites
Suggested Sites
Free Hotmail
Web Slice Gallery
KeepVid- Download and s...
Live Report ...
Practice ...
Page
Safety
Tools
Done
Computer | Protected Mode: Off
100%

Murphy's Laws

Original

If any can go wrong, it will.

Love Law

All the good ones are taken.

Computer Law

Any program, when running, is obsolete.

EMT Law

All bleeding stops...eventually.