CIS 4004: Web Based Information Technology Spring 2014

Inside HTML5 – Part 1

Instructor : Dr. Mark Llewellyn
markl@cs.ucf.edu
HEC 236, 407-823-2790
http://www.cs.ucf.edu/courses/cis4004/spr2014

Department of Electrical Engineering and Computer Science
University of Central Florida

Inside HTML5 – Part 1

- HTML5 has been created in a way to make it backward compatible with earlier versions of HTML and XHTML.

- HTML5 also tells the browser how it should handle errors caused by incorrect markup implementations. Previously, browsers would interpret the errors themselves, and each browser would have its own quirks.

- HTML5 has been written for developers and not just web page designers.

- The true power of HTML5, as we will see further into the course, is how it addresses the needs for web application developers. Because browsers are so powerful, websites can be created that are very much like applications. They can provide photo sharing, drawing, file editing, and other features.
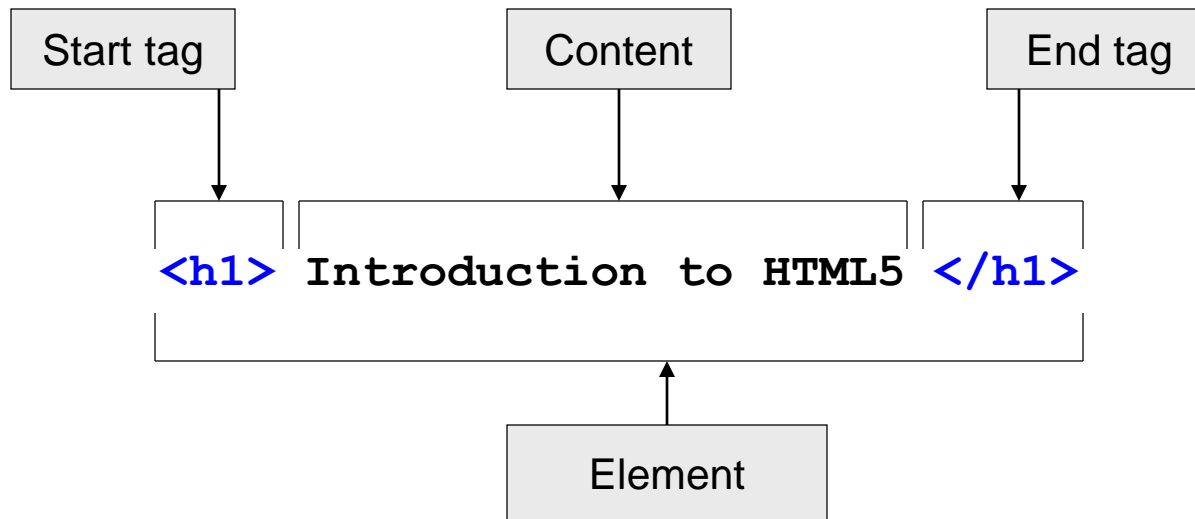
Inside HTML5 – Part 1

- Previously, these types of features required layers of JavaScript and a plug-in such as Java or Flash.

- This causes accessibility issues and relies heavily on the stability of third-party software.

- HTML5 provides new standards for how web applications can be created, with powerful APIs for this such as canvas drawing, drag and drop, offline storage, and native video in the browser.

- With specified standards, browsers will be able to handle these things correctly and in a stable fashion over time.

- As a quick review, the basics of an HTML element are shown on the next couple of pages.

Inside HTML5 – Part 1

| Start tag | | Content | | End tag |

```
<h1> Introduction to HTML5 </h1>
```

Element

- All elements, except for void (empty) elements, consist of three pieces: a start tag, content, and an end tag.

Inside HTML5 – Part 1

- Empty elements are used primarily to describe pieces of data that don't contain any content.

- For example, some common empty elements in HTML are:

    `<br>` for line break

    `<img>` for image

    `<p>` for paragraph

- In XML and HTML, 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 HTML. 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.

Inside HTML5 – Part 1

- HTML 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 appear in the start tag of an element.

- 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:
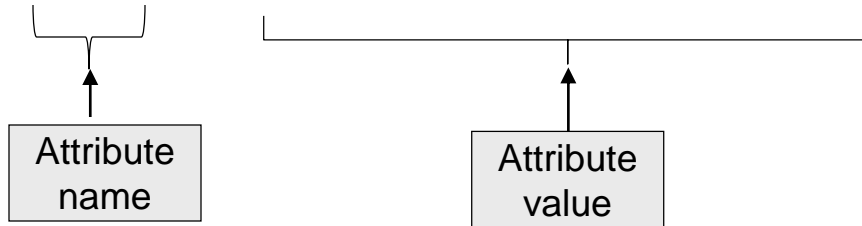
  ```
  <a href="http://mark.com">Click here</a>

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

Inside HTML5 – Part 1

```
<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
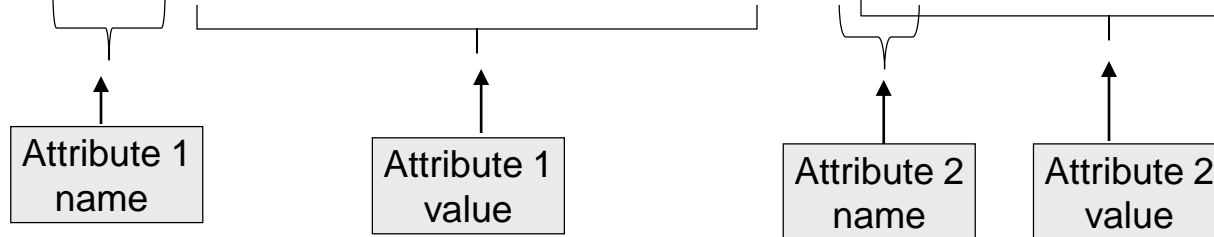
<img> empty element

Inside HTML5 – Part 1

- HTML5 is not just one technology. It is more of an umbrella term that has been adopted fro the inclusion of new and enhanced HTML elements, CSS styles, and JavaScript APIs and events.

- The intersection of these three technologies provides for a wide range of new features to enhance the user experience, make websites more like native applications, and integrate to devices.

- The following page lists just some of the new or enhanced functions available in HTML5:

New or Enhanced Functions In HTML5

- Improved semantics

- Forms

- Canvas drawing

- Drag and drop

- Local storage

- Page to page messaging

- Desktop notifications

- Video and audio

- Web sockets

- Geolocation (not strictly part of HTML5, but being developed in conjunction with it)

- History

- Microdata

The HTML5 logo. You can download this one and many other variants at http://w3.org/html/logo

New Structural Elements In HTML5

- HTML5 is not just about interactive creations with JavaScript APIs and video/audio coolness. There are more than 20 new elements designed to help you author your webpages, adding semantics to deliver more accessible, reusable content.

- Later on we'll examine new HTML5 form controls and multimedia elements, but for now we'll focus on the basics of creating a basic website template with the new elements.

- The basic page structure as defined by HTML5 appears on the next page.

Basic Page Structure - New HTML5 Elements

```
<header>
```

```
<nav>
```

```
<article>



    <section>



    <section>



    <section>
```

```
<aside>
```

```
<footer>
```

New Structural Elements In HTML5

- We'll do a quick review of the basic markup in an HTML5 document and clarify the differences from the older versions and what is and is not required in HTML5.

- The `doctype` should be the very first line in an HTML document. Called a Document Type Definition (DTD), the `doctype` is a web standard's requirement and it tells the browser how to process the document, which is why it must be the first thing in HTML document.

- The next page shows the old style XHTML `doctype` and the new HTML5 `doctype`.

New Structural Elements In HTML5

XHTML Strict 1.0 `doctype`:

```
<!DOCTYPE  html  PUBLIC  "-//W3C//DTD  XHTML  1.0  Strict//EN"
       "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

HTML5 `doctype`:

```
<!doctype html>
```

Historical Aside

If you go to: http://infomesh.net/html/history/early/ , you can see the earliest HTML document that can still be found on the Internet, from November 13, 1990. This document was created by Tim Berners-Lee.   The markup is quite simple, and its simplicity is the rationale behind the HTML5 standard. This markup contains a working hyperlink. In fact, if you copy this markup and add the new doctype to the page, it will validate against the HTML5 standard. Check it out!

New Structural Elements In HTML5

- The next thing you should do in all of your HTML documents is declare a language.

- Specifying the language of content is useful for a wide number of applications, from linguistically-sensitive searching to applying language-specific display properties.

- In some cases the potential applications for language information are still waiting for implementations to catch up, whereas in others, such as detection of language by voice browsers, it is a necessity today.

- Declaring a default language is important for applications such as accessibility (more later) and search engines.

New Structural Elements In HTML5

- The easiest way to specify the language of an HTML document is to add a `lang` attribute to the root element of the HTML page.

- The root element of an HTML page is always `<html>`, so to specify a language, you simply do the following:

```
<html lang="en">

<html lang="en-US">  //US dialect

<html lang="en-GB">  //British dialect
```

- Note that the lang attribute can also be used inline as shown:

```
<p>Elise replied <span lang="fr">

          "Bon chance!"</span></p>
```

New Structural Elements In HTML5

- Note that if you are familiar with XHTML, the `xmlns` attribute is no longer needed in HTML5.

        <html xmlns=http://www.w3.org/1999/xhtml>

- The `xmlns` attribute was used to inform the browser that all the elements on the page are in the XHTML namespace. However, in HTML5, elements are always in this namespace so it is no longer necessary to specify this attribute.

New Structural Elements In HTML5

- Next up in an HTML document is the `head` element.

- The first line you should include inside the `head` of an HTML5 document is the `charset` declaration, which tells the browser how the document should be interpreted; in this case, you want to send it an HTML document.

- In XHTML, this looked like this:

```
<meta http-equiv="Content-Type" content="text/html"; charset=utf-8">
```

- In HTML5, it is again, much simpler, and looks like this:

```
<meta charset="utf-8" />
```

New Structural Elements In HTML5

- The next line you should include inside the `head` of an HTML5 document is the `title` element, which tells the browser how the document should be titled in the browser header bar.

- This element is the same in HTML5 as it was in XHTML and looks like this:

    ```
    <title> Name of my page </title>
    ```

- Note that this is not a void element as its content is the title.

New Structural Elements In HTML5

- HTML5 helps you to reduce the amount of markup from your pages, and calls to JavaScript and CSS links are also reduced in size.

- In XHTML, they looked like this:

```
<script type="text/javascript" src="myscript.js"> </script>

<link rel="stylesheet" type="text/css" href="mystyles.css" />
```

- In HTML5, they're again much simpler, and look like this:

```
<script src="myscript.js"></script>

<link rel="stylesheet" href="mystyles.css" />
```

Syntax Writing Style In HTML5

- You can markup in either lowercase, or uppercase, or a combination of the two.

- You can markup with either quotation marks on attribute values, or omit quotation marks on attribute values, or a combination of the two.

- For void elements, such as a `<link>` element, you can omit the closing slash or include it.

- All below are equivalent:

```
<link rel="stylesheet" href="mystyles.css" />

<LINK REL="STYLESHEET" HREF="MYSTYLES.CSS" />

<liNK reL="styleSHEET" href=mystyles.css>
```

Syntax Writing Style In HTML5

- For backward compatibility, HTML5 allows web pages to be marked up using either HTML4 or XHTML syntax.

- While it is therefore possible to mark up your documents in HTML5 in either a very loose or rather strict style, I'd recommend using the more strict style. Why?

  - Rules are useful. They enable collaborative working, allowing everyone to conform to a standardized syntax.

  - The rules also make it easier to learn markup.

  - Most importantly, accessibility best practices will often require it. For example, the specification of a language.

File   Edit   Search   View   Encoding   Language   Settings   Macro   Run   Plugins   Window   ?   X

notes.html ☒   homework.html ☒   index.html ☒   homework.html ☒   html5-template-advanced.html ☒   html5-template-minima ◄ ►

```
 1    <!DOCTYPE html>
 2    <html lang="en">
 3    <head>
 4        <meta charset="utf-8" />
 5        <title> Page Title </title>
 6        <link rel="stylesheet" href="mystylefile.css"  />
 7    </head>
 8    <body>
 9        <!-- page content goes here -->
10    </body>
11    </html>
```

A minimal HTML template

length : 219   lines : 11   Ln : 1   Col : 1   Sel : 0 | 0   Dos\Windows   ANSI as UTF-8   INS

File   Edit   Search   View   Encoding   Language   Settings   Macro   Run   Plugins   Window   ?   X

notes.html ✕ | homework.html ✕ | index.html ✕ | homework.html ✕ | html5-template-advanced.html ✕

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8" />
    <title> Page Title </title>
    <link rel="stylesheet" href="mystylefile.css"  />
</head>
<body>
    <!-- page content goes here -->
    <header>
        <!-- header content goes here -->
    </header>
    <nav>
        <!-- navigational elements goes here -->
    </nav>
    <article>
        <!-- main article content goes here -->
        <section>
            <!-- sectional content goes here  - possibly several -->
        </section>
    </article>
    <aside>
        <!-- supplemental content goes here -->
    </aside>
    <footer>
        <!-- footer content goes here -->
    </footer>
</body>
</html>
```

A more complete HTML template

Hyper Text Markup Language file   length : 620   lines : 29   Ln : 1   Col : 16   Sel : 0 | 0   Dos\Windows   ANSI as UTF-8   INS

New Structural Elements In HTML5

- In 2005, Google analyzed more than 1 billion web pages to find out what class names were being used by developers and web authors. This provided the basis for the new elements in HTML5.

- The following are the 20 most popular class names at the time:

footer	menu	Title	Small	Text
Content	Header	Nav	Copyright	Button
Main	Search	Msonormal	Date	Smalltext
Body	Style1	Top	White	link

- Although several of these items are presentational (for example, `white, style1, msnormal`), the others make up the elements included in the HTML5 specification.

New Structural Elements In HTML5

- So why use these new elements?

- HTML5 allows you to give your content semantic meaning, so, for example, if you have navigation on your page, you can use the `nav` element because that element provides meaning to its content.

- As an example, let's start at the top of a "typical" web page. The header element is often the first thing on a web page, and it usually contains things like a logo, the website name, or the main site navigation.

- Things like a search form or a table of contents are also often included in the header element.

New Structural Elements In HTML5

- The HTML5 specification states that the `<header>` element can contain navigation aids, but it is also possible for the navigation aids to be placed outside of the `<header>`.

- You are also not restricted to just a single `<header>` elements per page, and it does not have to be at the top of a page.

- You cannot place a `<header>` inside a `<footer>`, `<address>`, or another `<header>` element.

- The markup example on the next page illustrates this concept.

CAUTION!!

Do not confuse the <header> element with the <head> element.
There can only be one <head> element in each HTML5 document.

File   Edit   Search   View   Encoding   Language   Settings   Macro   Run   Plugins   Window   ?   X

homework.html   html5-template-advanced.html   html5-template-minimal.html   multiple headers on one page.html

```
 1  <!DOCTYPE html>
 2  <html lang="en">
 3  <head>
 4      <meta charset="utf-8" />
 5      <title> Using Multiple Headers On One Page </title>
 6  </head>
 7  <body>
 8      <article>
 9          <header>
10              <h1><a href="#">Chapter 1</a></h1>
11              <p>1.7.2014</p>
12          </header>
13          <p>This is the text for the first chapter.
14             It could go on for a long way.
15          </p>
16      </article>
17      <article>
18          <header>          |
19              <h1><a href="#">Chapter 2</a></h1>
20              <p>1.6.2014</p>
21          </header>
22          <p> This is the text for the second chapter.
23             It too, could go on for a long way.
24          </p>
25      </article>
26      <article>
27          <header>
28              <h1><a href="#">Chapter 3</a></h1>
29              <p>1.7.2014</p>
30              <p>This is the text for the third chapter.
31              </p>
32          </header>
33      </article>
34  </body>
35  </html>
```

length : 728   lines : 35   Ln : 18   Col : 21   Sel : 0 | 0   Dos\Windows   ANSI as UTF-8   INS

Chapter 1

1.7.2014

This is the text for the first chapter. It could go on for a long way.

Chapter 2

1.6.2014

This is the text for the second chapter. It too, could go on for a long way.

Chapter 3

1.7.2014

This is the text for the third chapter.

Grouping Headings With The `<hgroup>` Element

- The `<hgroup>` element allows you to add further information to your `<header>` elements.

- This element is used to group more than one related `h1-6` headings. It can contain only `<h1>`...`<h6>` elements. If your site has a subheading or a tagline, you would use this element to group them together.

- Although it is a useful grouping option, the `<hgroup>` element is primarily intended to tell the document outline (we'll learn about this later) which of the headings is the most important. In the example, on the next page, the document outline will exclude all headings within the `hgroup` except the highest one, which in this case, is the `h1` heading.

- As you can see in the markup, the `h2` heading is relevant to the `h1` heading, and so on, so the use of the `hgroup` makes sense. If you have a single heading element, there would be no need to use the `hgroup`.

File   Edit   Search   View   Encoding   Language   Settings   Macro   Run   Plugins

Opera

html5-template-advanced.html ☒   html5-template-minimal.html ☒   multiple headers on one

file:///C:/COURSES/CIS%204004%20-%20Web%20Based%20Information9

```
 1    <!DOCTYPE html>
 2    <html lang="en">
 3    <head>
 4        <meta charset="utf-8" />
 5        <title> Using the hgroup element </title>
 6    </head>
 7    <body>
 8        <header>
 9            <hgroup>
10                <h1><a href="#"> Inside HTML5 - Part 1</a></h1>
11                <h2>An in-depth look at the new structural elements of HTML5</h2>
12                <h3>Really cool features of HTML5</h3>
13            </hgroup>
14        </header>
15    </body>
16    </html>
```

Inside HTML5 - Part 1

An in-depth look at the new structural elements of HTML5

Really cool features of HTML5

Hyper Text Markup L   length : 360   lines : 16   Ln : 1   Col : 1   Sel : 0 | 0   Dos\Windows   ANSI as UTF-8   INS

Creating Navigation With The `<nav>` Element

- The `<nav>` element, as you might expect from it name, is for navigational content. It is used to link to other pages within the site or to other parts of the page, like a table of contents, for example.

- The most common use of a `<nav>` element is for the main navigation on a website. It is common practice to use an unordered list to code navigation content.

- The following page illustrates both the traditional way of marking up navigation and the way to do this in HTML5. Notice that there isn't much difference between the two, but it is profound from a semantic point of view.

Creating Navigation With The `<nav>` Element

```
<ul id="nav">
    <li><a href="#">Home</a> </li>
    <li><a href="#">About</a> </li>
    <li><a href="#">News</a> </li>
    <li><a href="#">Contact Us</a> </li>
</ul>
```

Traditional Way of Marking Up Navigational Content

```
<nav>
    <ul>
        <li><a href="#">Home</a> </li>
        <li><a href="#">About</a> </li>
        <li><a href="#">News</a> </li>
        <li><a href="#">Contact Us</a> </li>
    </ul>
</nav>
```

Navigational Content Markup In HTML5 Using `<nav>` Element

Creating Navigation With The `<nav>` Element

- As the example on the next page illustrates, the `<nav>` element can also be placed in the `<header>` element. In HTML5 the `<header>` element allows for introductory and navigational content.

- Navigational content does not have to be placed in a `<header>` element, and sometimes its placement might depend more on styling issues.

- It is quite common to see a navigation menu in the footer of a page, sometimes duplicating the main site navigation.

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

html5-template-minimal.html | multiple headers on one page.html |

```
 1  <!DOCTYPE html>
 2  <html lang="en">
 3  <head>
 4      <meta charset="utf-8" />
 5      <title> Navigation element insid
 6      <style>
 7      <!--
 8          li {display: inline; }
 9      -->
10      </style>
11  </head>
12  <body>
13      <!-- content goes here -->
14      <header>
15          <h1>My Super HTML5 Site</h1>
16          <nav>
17              <ul>
18                  <li><a href="#Home">Home</a>    </li>
19                  <li><a href="#About">About</a>    </li>
20                  <li><a href="#News">News</a>   </li>
21                  <li><a href="#ContactUs">Contact Us</a> </li>
22              </ul>
23          </nav>
24      </header>
25      <hr />
26      <h3> Headline #1 </h3>
27      <p> Headline #1 content goes here. </p>
28  </body>
29  </html>
```

Opera

file:///C:/COURSES/CIS%204004%20-%20W

My Super HTML5 Site

Home   About   News   Contact Us

Headline #1

Headline #1 content goes here.

Hyper Text Markup L: length : 614   lines : 29   Ln : 1   Col : 1   Sel : 0 | 0   Dos\Windows   ANSI as UTF-8   INS

Creating Navigation With The `<nav>` Element

- There is a big accessibility win when using the `<nav>` element. Assistive technologies such as screen readers, will be able to search and immediately use groups of navigation rather than waiting for them to appear on screen.

- Traditionally, developers have used "skip" or "jump" links of the very first things in an HTML document, and they are usually links to other main navigation or main content.

- However, using the `<nav>` element means that developers will be able to drop such "skip" menus.

- The only problem at the moment is that most assistive technologies have limited support for HTML5 elements.. However, they will soon catch up.

The New `<article>` Element

- The `<article>` element and the `<section>` element (discussed later) are arguably the two most important new HTML5 structural elements. However, they are also two of the more confusing new elements.

- The `<article>` element is an independent block of content; it is content that could exist in its own right and content that is reusable.

- Consider content you see in an RSS feed; the content is nearly always individual articles. You could take them out of the feed, and they make sense on their own.

- The HTML5 specification suggests some examples of how an article element can be used, such as a forum post, a magazine or newspaper article, a blog entry, or a user-submitted comment.

The New `<section>` Element

- The `<section>` element is an area of content or an area of a page that nearly always requires a heading. It should be used to group an entire "section" of content. A section can be broken down into further sections if required.

- It is **not** to be used as a general wrapper for styling purposes (i.e., a `<div>` element).

- A `<section>` element can contain `<article>` elements, and `<article>` elements can have their content split into multiple `<section>` elements.

- Before we look at an example using these new elements, we'll consider a couple of related new structural element, the `<aside>` element and the `<footer>` element.

The `<aside>` Element

- The `<aside>` element is for a group of content that is "tangentially" related to its surrounding content, such as a list of most popular posts, blog categories, or recent comments. This type of content is related to the main page content, but it is also separate from it.

- In current web development, it is common for there to be a "sidebar" on the page. This does not necessarily mean that it is physically on the side of the page, but it often contains things such as related links or a list of categories.

- The correct use of the `<aside>` element depends on where you put it. If it is inside an article, the `<aside>` content should tangentially relate to the article content, such as a glossary. If the `<aside>` is is outside of an article or section, its content must be related to the page, such as related links, the site owner's Twitter feed, etc.

The `<footer>` Element

- The `<footer>` element, as its name suggests, is typically located at the bottom of the page. However, that is not always the case, although the it will often be found there.

- The `<footer>` element is intended for content about its section, including information about the author or site owner, copyright data, and site terms and conditions.

- If a `<footer>` element is inside an `<article>` or `<section>`, it could contain the date the article was published, tags, categories, and other metadata.

- The HTML5 specification suggests a solution to a very common web element: the copyright notice on a page:
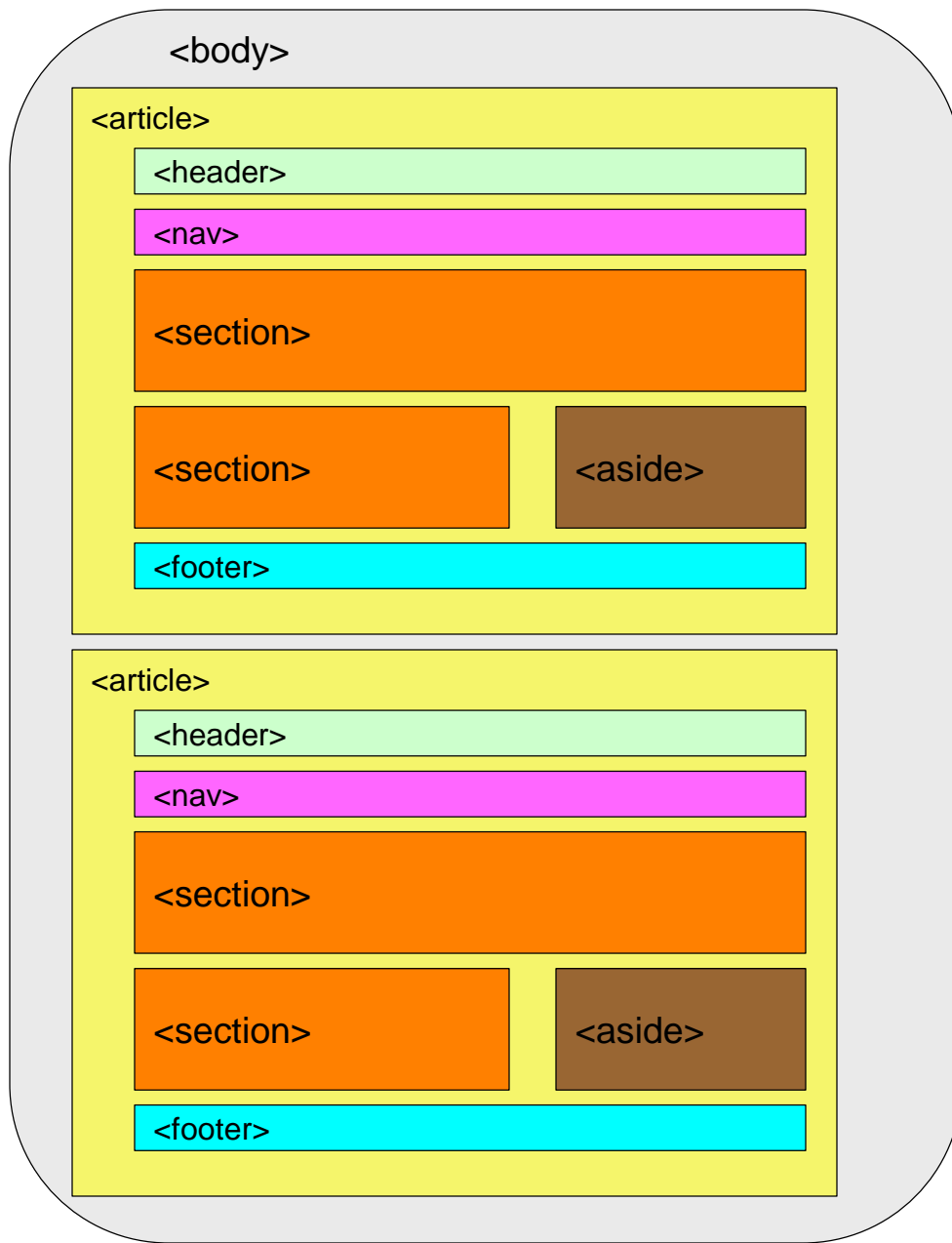
```
<footer> <small>&copy; Copyright MJL 2012</small> </footer>
```

The `<footer>` Element

- The previous example of the copyright would most likely be located just before the closing `</body>` tag. Notice how the use of the <small> tag is recommended.

- Like the `<header>` element, you can use a `<footer>` element more than once per page.

- You can put a `<footer>` element inside an `<article>` element.

- The following example builds a page with many aspects of the `<article>`, `<section>`, `<aside>`, and `<footer>` elements utilized.

<body>

<article>
<header>

<nav>

<section>

<section>   <aside>

<footer>

<article>
<header>

<nav>

<section>

<section>   <aside>

<footer>

An HTML5 page
illustrating the use of
`header`, `nav`,
`section`, `aside`,
and `footer`
elements.

A Big Example

- The next few pages present an example webpage that utilizes all of the new HTML5 structural elements that were presented in this set of notes.

- The page that is created represents a layout for a news page. It includes some basic CSS to position the various elements, but I didn't include anything too advanced as far as the CSS is concerned; we'll get more into that later.

- Since the markup is fairly long, I've abbreviated some of the more repetitive parts here. The full markup is available on the course website.

File   Edit   Search   View   Encoding   Language   Settings   Macro   Run   Plugins   Window   ?

multiple headers on one page.html | hgroup example.html | navigation inside header.html | newssite - notes version.html | newssite.html

```html
1   <!DOCTYPE html>
2   <html lang="en">
3   <head>
4       <meta charset="utf-8">
5       <title>Loadsa News - the best news site there ever was!</title>
6       <style>
7       <!--
8           header, nav, section, article, footer {display: block;}
9           header, nav {border-bottom: 1px dotted #000; clear: both; width: 100%;}
10          nav li {display: inline;}
11          section#headline {clear: both; border: 5px solid #000; padding: 1%; width: 97%;}
12          section#sports, section#entertainment, section#tech {float: left; margin: 0 5px; padding: 1%; width:
13          aside, footer {clear: both;}
14          aside img {border: 1px solid #ccc; margin: 0 10px 0 0;}
15      -->
16      </style>
17  </head>
18  <body>
19
20  <header>
21      <hgroup>
22          <h1>Loadsa News</h1>
23          <h2>Bringing you all kinds of news!</h2>
24      </hgroup>
25  </header>
26
```

Hyper Text Markup Language file   length : 2899   lines : 97   Ln : 14   Col : 56   Sel : 0 | 0   Dos\Windows   ANSI as UTF-8   INS

File   Edit   Search   View   Encoding   Language   Settings   Macro   Run   Plugins   Window   ?   X

multiple headers on one page.html | hgroup example.html | navigation inside header.html | newssite - notes version.html | newssite.html

```
27  <nav>
28      <ul>
29          <li><a href="#">Home</a></li>  
30          <li><a href="#">Sports news</a></li>  
31          <li><a href="#">Entertainment news</a></li>  
32          <li><a href="#">Tech news</a></li>  
33          <li><a href="#">About</a></li>  
34          <li><a href="#">Contact</a></li>
35      </ul>
36  </nav>
37
38  <section id="headline">
39      <h1>Headline article</h1>
40      <article>
41          <header>
42          <h2><a href="#">This is our most important article</a></h2>
43          <p>1st January 2014</p>
44          </header>
45          <p>Chris Froome targets win in 2014 Tour de France.</p>
46      </article>
47  </section>
48
```
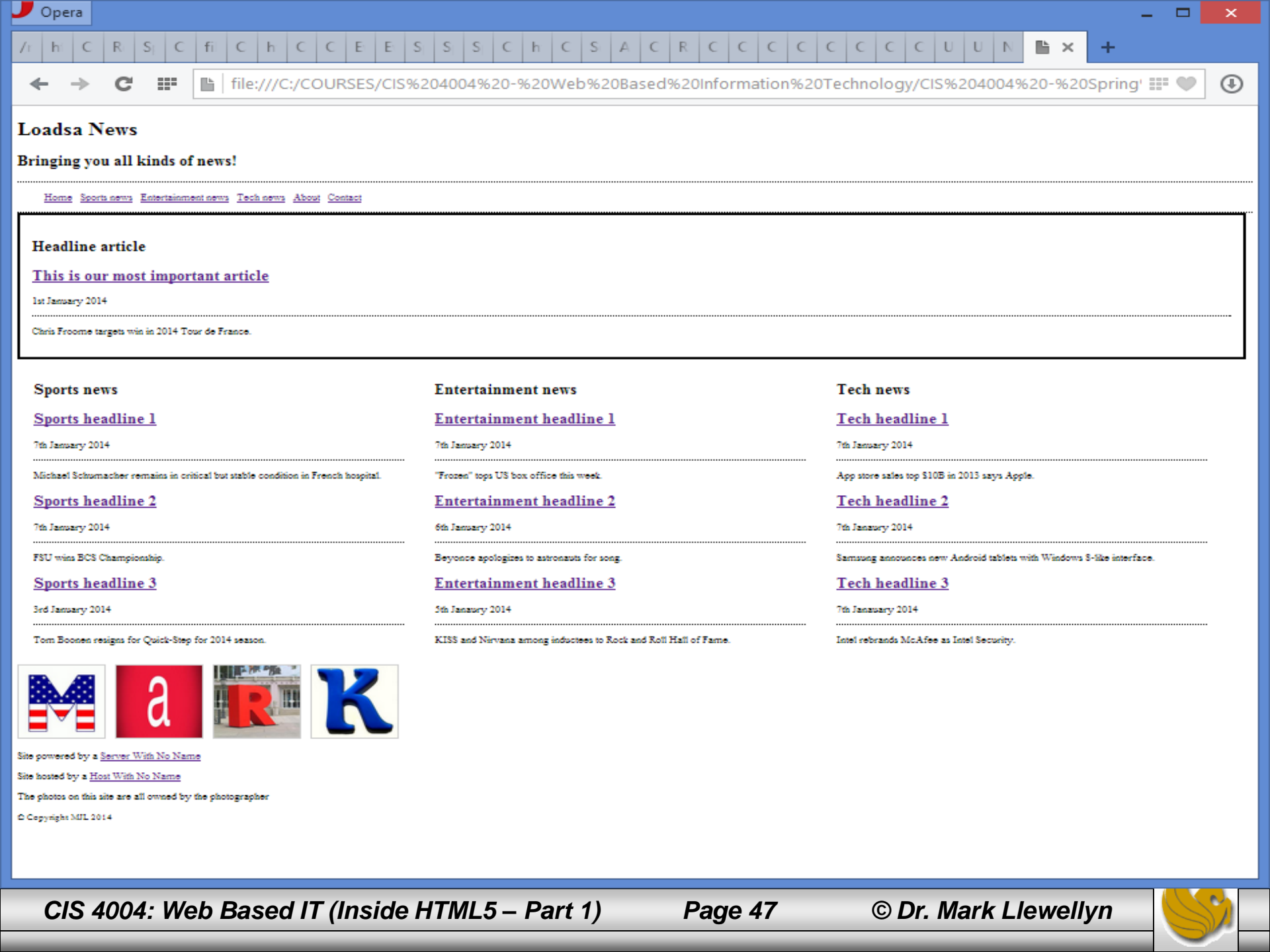
Hyper Text Markup Language file   length : 3871   lines : 140   Ln : 120   Col : 56   Sel : 0 | 0   Dos\Windows   ANSI as UTF-8   INS

File   Edit   Search   View   Encoding   Language   Settings   Macro   Run   Plugins   Window   ?   X

multiple headers on one page.html ✕ | hgroup example.html ✕ | navigation inside header.html ✕ | newssite - notes version.html ✕ | newssite.html ✕

```
48
49  <section id="sports">
50      <h1>Sports news</h1>
51      <article>
52          <header>
53          <h2><a href="#">Sports headline 1</a></h2>
54          <p>7th January 2014</p>
55          </header>
56          <p>Michael Schumacher remains in critical but stable condition in French hospital.</p>
57      </article>
58      <article>
59          <header>
60          <h2><a href="#">Sports headline 2</a></h2>
61          <p>7th January 2014</p>
62          </header>
63          <p>FSU wins BCS Championship.</p>
64      </article>
65      <article>
66          <header>
67          <h2><a href="#">Sports headline 3</a></h2>
68          <p>3rd January 2014</p>
69          </header>
70          <p>Tom Boonen resigns for Quick-Step for 2014 season.</p>
71      </article>
72  </section>
```

Hyper Text Markup Language file   length : 3871   lines : 140   Ln : 120   Col : 56   Sel : 0 | 0   Dos\Windows   ANSI as UTF-8   INS

File   Edit   Search   View   Encoding   Language   Settings   Macro   Run   Plugins   Window   ?   X

multiple headers on one page.html ☒ | hgroup example.html ☒ | navigation inside header.html ☒ | newssite - notes version.html ☒ | newssite.html ☒

```
116        <header>
117        <h2><a href="#">Tech headline 3</a></h2>
118        <p>7th Janauary 2014</p>
119        </header>
120        <p>Intel rebrands McAfee as Intel Security.</p>
121    </article>
122  </section>
123
124  <aside>
125    <a href="#"><img alt="Snazzy advert" src="images/the-letter-m.jpg" height="128" width="128" /></a
126    <a href="#"><img alt="Snazzy advert" src="images/the-letter-a.jpg" height="128" width="128" /></a
127    <a href="#"><img alt="Snazzy advert" src="images/the-letter-r.jpg" height="128" width="128" /></a
128    <a href="#"><img alt="Snazzy advert" src="images/the-letter-k.jpg" height="128" width="128" /></a
129  </aside>
130
131
132  <footer>
133    <p>Site powered by a <a href="#">Server With No Name</a></p>
134    <p>Site hosted by a <a href="#">Host With No Name</a></p>
135    <p>The photos on this site are all owned by the photographer</p>
136    <small>&copy; Copyright MJL 2014</small>
137  </footer>
138
139  </body>
140  </html>
```

Hyper Text Markup Language file   length : 3871   lines : 140   Ln : 120   Col : 56   Sel : 0 | 0   Dos\Windows   ANSI as UTF-8   INS

Loadsa News

Bringing you all kinds of news!

Home   Sports news   Entertainment news   Tech news   About   Contact

---

Headline article

This is our most important article

1st January 2014

Chris Froome targets win in 2014 Tour de France.

Sports news

Sports headline 1

7th January 2014

Michael Schumacher remains in critical but stable condition in French hospital.

Sports headline 2

7th January 2014

FSU wins BCS Championship.

Sports headline 3

3rd January 2014

Tom Boonen resigns for Quick-Step for 2014 season.

Entertainment news

Entertainment headline 1

7th January 2014

"Frozen" tops US box office this week.

Entertainment headline 2

6th January 2014

Beyonce apologizes to astronauts for song.

Entertainment headline 3

5th January 2014

KISS and Nirvana among inductees to Rock and Roll Hall of Fame.

Tech news

Tech headline 1

7th January 2014

App store sales top $10B in 2013 says Apple.

Tech headline 2

7th January 2014

Samsung announces new Android tablets with Windows 8-like interface.

Tech headline 3

7th January 2014

Intel rebrands McAfee as Intel Security.

Site powered by a Server With No Name

Site hosted by a Host With No Name

The photos on this site are all owned by the photographer

© Copyright MJL 2014

Using The HTML5 Outliner To Ensure Correct Structure

- With the new HTML5 structural elements, you have the opportunity to make your content flow in a logical manner and to allow people to navigate through the content using a hierarchy (using screen readers, for example), rather like a table of contents.

- Testing against the outline will allow you to check that you are using headings and sections correctly.

- There are various browser extensions and websites at your disposal, but here I'll show you how to get and use a Google Chrome extension.

- Download the Chrome extension at: http://code.google.com/p/h5o.

- Once the extension is installed, you'll get an icon in the address bar as shown on the next page.

Using The HTML5 Outliner To Ensure Correct Structure

The outliner icon

Using The HTML5 Outliner To Ensure Correct Structure

- When you select this tool, you will see displayed data that looks like a table of contents, usually with the content indented.

- If you have organized the content properly, you should have a structured and logical table of contents.

- You want to avoid "Untitled section/article." If that message is displayed, chances are that you have used the wrong markup, and you should reexamine your markup and correct the issues.

- Note, however, that `nav` and `aside` are allowed to have "Untitled section."

- The next three pages illustrate the use of the outliner tool. The last of these pages shows the outline of our big example news page.

EDITION: INTERNATIONAL   U.S.   MÉXICO   ARABIC

Sign up | Log in

TV: CNNi   CNN en Español

Set edition preference

SEARCH

POWERED BY Google

Home   Video   World   U.S.   Africa   Asia   Travel   iReport

January 7, 2014 – Updated 1818 GMT (0218 HKT) Edited by D

EDITOR'S CHOICE   Scenes from the field

1. Angry Dennis Rodman defends N. Korea game
   1. EXTREME WEATHER
   2. LATEST NEWS
   3. FIVE STORIES NOT TO MISS
   4. OPINION AND ANALYSIS
2. What has Kerry achieved in Mideast?
   1. CNN TV
      1. BUSINESS
      2. WORLD SPORT
      3. DEFINING MOMENTS
   2. Baby giant panda
      1. MOST POPULAR
      2. WATCH THESE VIDEOS
   3. Plane struggles in wind
      1. TRAVEL
      2. STYLE
   4. 3-D printed masterpieces
      1. Popular on Facebook
      2. ON CHINA
   5. China crushes tons of illegal ivory
      1. iREPORT
      2. AMANPOUR
      3. Quick Vote
         1. Do you suffer from stage fright?

EXCLUSIVE

Angry Dennis Rodman
defends N. Korea game

Basketball star Dennis Rodman
defends his visit to North Korea with
a team of former NBA players, saying
it was a "great idea for the world."

FULL STORY

• Rodman to CNN: I don't give a s***
• Dennis Rodman: I love my friend

edition.cnn.com/SPORT/

U.S. Secretary of State John Kerry has made 10 trips to the
Mideast in the past year in search of the elusive prize of a peace
deal between Israel and the Palestinians. Is he any closer?

ICE CHAOS

polar vortex

• Janet Yellin

ADVERTISEMENT

FOLLOW US

Like 3.3m   Follow @cnni

CNN TV   Royal Television Society
'News channel of the year'

Featured TV

Apps   Zimbra Web Client L...   Cycling News & Rac...   News 13 - Orlando ...   Imported From IE

Web Images Videos Maps News Shopping more ▼   Tue, Jan 7 | 1:47 PM

1. *Untitled BODY*

[   ] Search

Recommended Sites:

Facebook   Free Games   Amazon

Chat Now   YouTube   eBay

More Languages ▼

©2013 Bandoo Media Inc.   Privacy

eustis.eecs.ucf.edu/~markl/newssite.html

Apps | Zimbra Web Client L... | Cycling News & Rac... | News 13 - Orlando ... | Imported From IE

Loadsa News

Bringing you all kinds of news!

Home   Sports news   Entertainment news   Tech news   About   Contact

1. Loadsa News
   1. *Untitled NAV*
   2. Headline article
      1. This is our most important article
   3. Sports news
      1. Sports headline 1
      2. Sports headline 2
      3. Sports headline 3
   4. Entertainment news
      1. Entertainment headline 1
      2. Entertainment headline 2
      3. Entertainment headline 3
   5. Tech news
      1. Tech headline 1
      2. Tech headline 2
      3. Tech headline 3
   6. *Untitled ASIDE*

Headline article

This is our most important a

1st January 2014

Chris Froome targets win in 2014 Tour de Fran

Sports news

Sports headline 1

7th January 2014

Entertainment news

Entertainment headline 1

7th January 2014

Tech news

Tech headline 1

7th January 2014