

Introduction to HTML

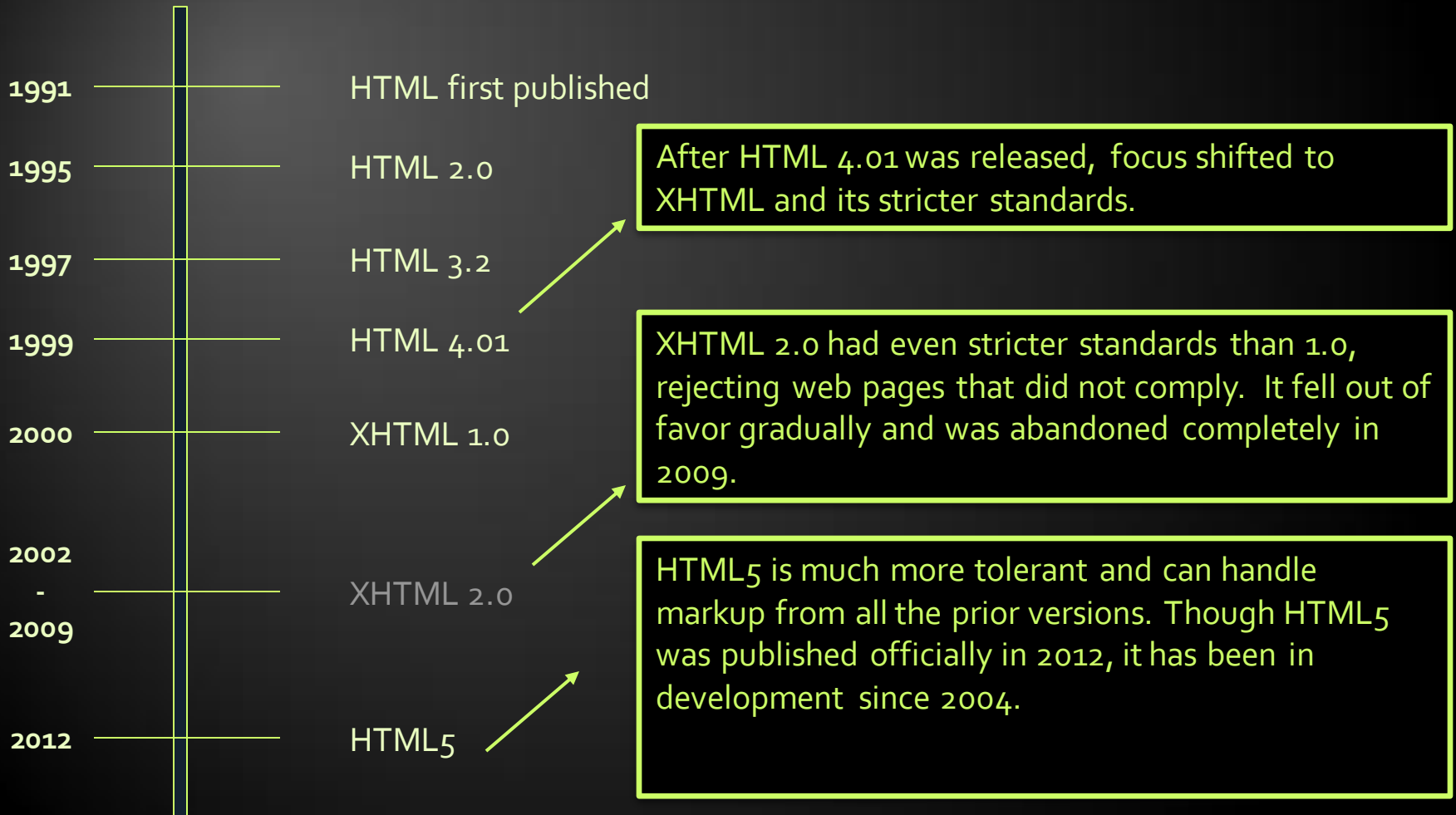
Course Code: CIS121

Module Leader: Md. Minhaj Hosen

Lecturer (CIS),

Daffodil International University (DIU)

History of HTML

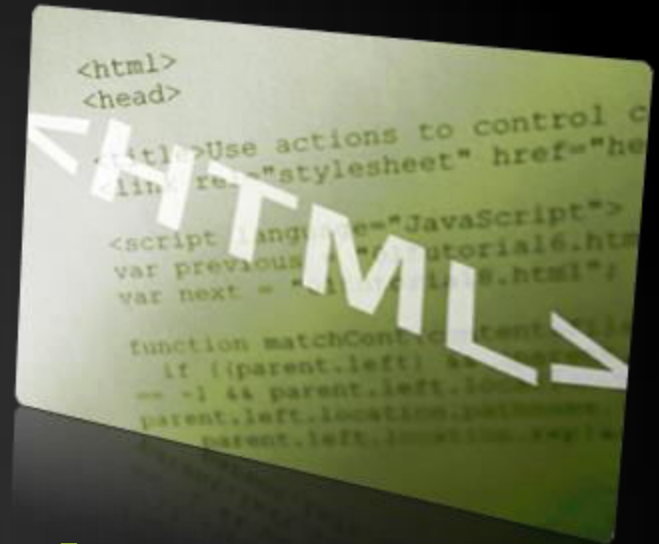


What is a Web Page?

- ◆ Web pages are text files containing HTML
- ◆ HTML – Hyper Text Markup Language
 - ◆ A notation for describing
 - ◆ document structure (semantic markup)
 - ◆ formatting (presentation markup)
 - ◆ Looks (looked?) like:
 - ◆ A Microsoft Word document
- ◆ The markup tags provide information about the page content structure

Creating HTML Pages

- ◆ An HTML file must have an `.htm` or `.html` file extension
- ◆ HTML files can be created with text editors:
 - ◆ NotePad, NotePad ++, PSPad
- ◆ Or HTML editors (WYSIWYG Editors):
 - ◆ Microsoft FrontPage
 - ◆ Macromedia Dreamweaver
 - ◆ Netscape Composer
 - ◆ Microsoft Word
 - ◆ Visual Studio



Text, Images, Tables, Forms



HTML Structure

- ◆ HTML is comprised of “elements” and “tags”

- ◆ Begins with `<html>` and ends with `</html>`

- ◆ Elements (tags) are nested one inside another:

```
<html> <head></head> <body></body> </html>
```

- ◆ Tags have attributes:

```

```

- ◆ HTML describes structure using two main sections:
`<head>` and `<body>`

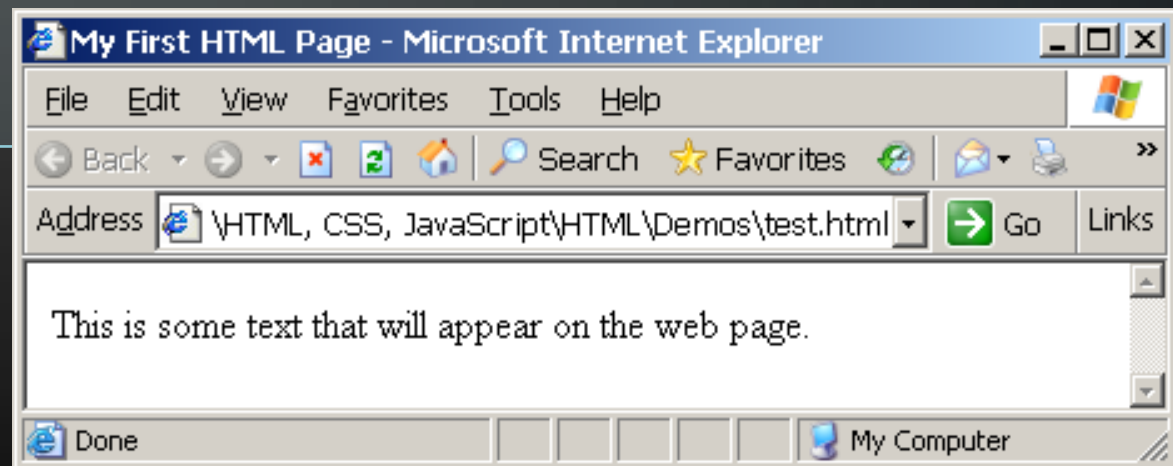
HTML Code Formatting

- ◆ The HTML source code should be formatted to increase readability and facilitate debugging.
 - ◆ Every block element should start on a new line.
 - ◆ Every nested (block) element should be indented.
 - ◆ Browsers ignore multiple whitespaces in the page source, so formatting is harmless.
- ◆ For performance reasons, formatting can be sacrificed

First HTML Page

test.html

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>My First HTML Page</title>
  </head>
  <body>
    <p>This is some text...</p>
  </body>
</html>
```



First HTML Page: Tags

```
<!DOCTYPE HTML>  
<html>  
  <head>  
    <title>My First HTML Page</title>  
  </head>  
  <body>  
    <p>This is some text...</p>  
  </body>  
</html>
```

Opening tag

Closing tag

An HTML element consists of an opening tag, a closing tag and the content inside.

First HTML Page: Header

HTML header

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>My First HTML Page</title>
  </head>
  <body>
    <p>This is some text...</p>
  </body>
</html>
```

First HTML Page: Body

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>My First HTML Page</title>
  </head>
  <body>
    <p>This is some text...</p>
  </body>
</html>
```



HTML body

Some Simple Tags

◆ HyperlinkTags

```
<a href="http://www.telerik.com/"  
  title="Telerik">Link to Telerik Web site</a>
```

◆ ImageTags

```

```

◆ Text formatting tags

```
This text is <em>emphasized.</em>  
<br />new line<br />  
This one is <strong>more emphasized.</strong>
```

Some Simple Tags – Example

some-tags.html

```
<!DOCTYPE HTML>
<html>
<head>
  <title>Simple Tags Demo</title>
</head>
<body>
<a href="http://www.telerik.com/" title=
  "Telerik site">This is a link.</a>
<br />

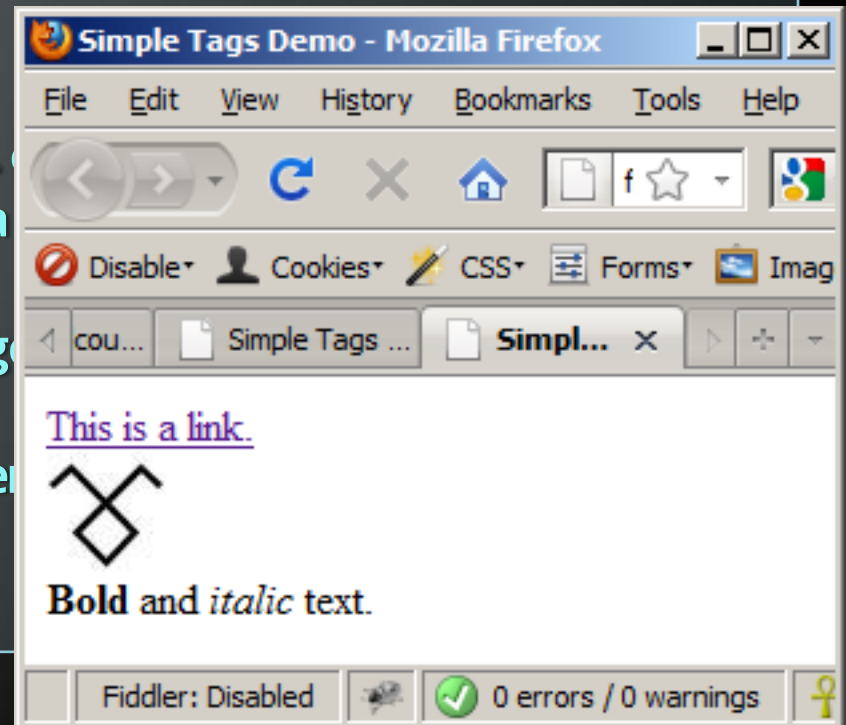
<br />
<strong>Bold</strong> and <em>italic</em> text.
</body>
</html>
```

Some Simple Tags – Example (2)

some-tags.html

```
<!DOCTYPE HTML>
<html>
<head>
  <title>Simple Tags Demo</title>
</head>
<body>
<a href="http://www.telerik.com"
  "Telerik site">This is a
<br />

<br />
<strong>Bold</strong> and <em>italic</em>
</body>
</html>
```



Tags Attributes

- ◆ Tags can have attributes

- ◆ Attributes specify properties and behavior

- ◆ Example:

Attribute `alt` with value "logo"

```

```

- ◆ Few attributes can apply to every element:

- ◆ `id`, `style`, `class`, `title`
- ◆ The `id` is unique in the document
- ◆ Content of `title` attribute is displayed as hint when the element is hovered with the mouse
- ◆ Some elements have obligatory attributes

Headings and Paragraphs

◆ Heading Tags (h1 – h6)

```
<h1>Heading 1</h1>  
<h2>Sub heading 2</h2>  
<h3>Sub heading 3</h3>
```

◆ Paragraph Tags

```
<p>This is my first paragraph</p>  
<p>This is my second paragraph</p>
```

◆ Sections: div and span

```
<div style="background: skyblue;">  
  This is a div</div>
```


Headings and Paragraphs – Example

headings.html

```
<!DOCTYPE HTML>
<html>
  <head><title>Headings and paragraphs</title></head>
  <body>
    <h1>Heading 1</h1>
    <h2>Sub heading 2</h2>
    <h3>Sub heading 3</h3>

    <p>This is my first paragraph</p>
    <p>This is my second paragraph</p>

    <div style="background:skyblue">
      This is a div</div>
  </body>
</html>
```

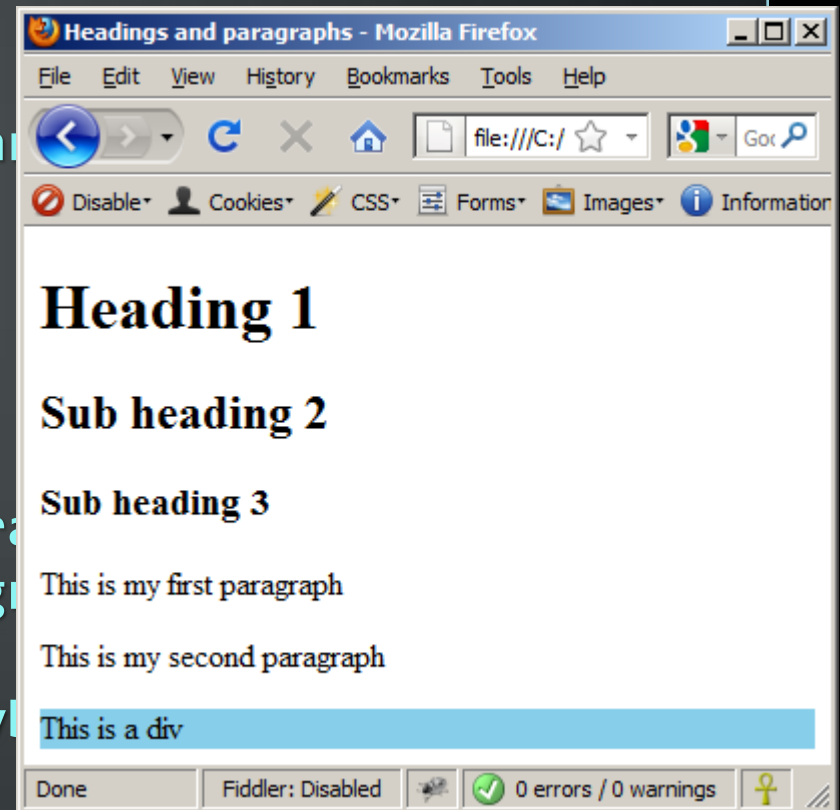
Headings and Paragraphs – Example (2)

headings.html

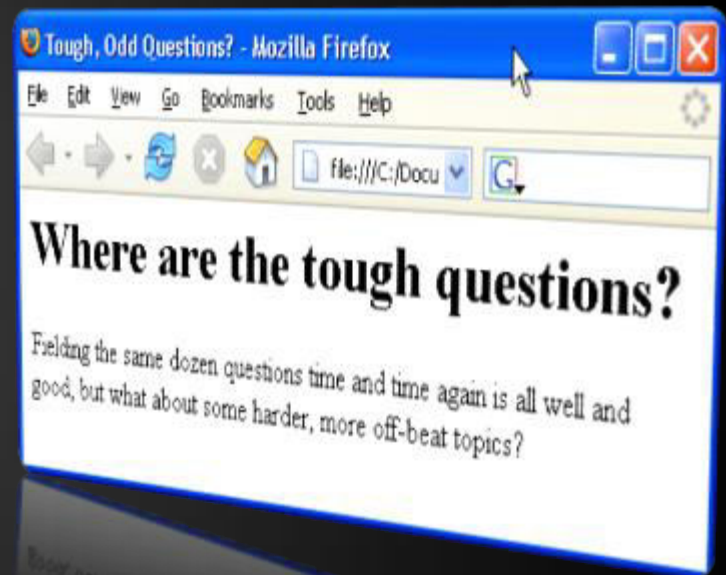
```
<!DOCTYPE HTML>
<html>
  <head><title>Headings and paragraphs</title>
  <body>
    <h1>Heading 1</h1>
    <h2>Sub heading 2</h2>
    <h3>Sub heading 3</h3>

    <p>This is my first paragraph</p>
    <p>This is my second paragraph</p>

    <div style="background:skyblue">
      This is a div</div>
  </body>
</html>
```



```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
2 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml">
4 <head>
5   <title>Tabview - Demo</title>
6
7   <script src="prototype.js" type="text/javascript"></script>
8   <script src="tabview.js" type="text/javascript"></script>
9
10  <link href="tabview.css" rel="stylesheet" type="text/css" />
11 </head>
12 <body id="body">
13
14   <ul class="tab-collection">
15     <li class="tab" title="Tab1">
16       <h1>Tab 1</h1>
17       
18     </li>
19
20     <li class="tab" title="Tab2">
21       <h1>Tab 2</h1>
22       
23     </li>
24
25     <li class="tab" title="Tab3">
26       <h1>Tab 3</h1>
27       
28     </li>
29   </ul>
30
31   <script type="text/javascript">
32     UI.Tabview.init( 'body', 1, width: 600px);
33   </script>
34 </body>
35 </html>
```



Introduction to HTML

HTML Document Structure in Depth

The <!DOCTYPE> Declaration

- ◆ HTML documents must start with a document type definition (DTD)
 - ◆ It tells web browsers what type is the served code
 - ◆ Possible versions: HTML 4.01, XHTML 1.0 (Transitional or Strict), XHTML 1.1, HTML 5

- ◆ Example:

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

- ◆ See <http://w3.org/QA/2002/04/valid-dtd-list.html> for a list of possible doctypes

HTML vs. XHTML



- ◆ XHTML is more strict than HTML
 - ◆ Tags and attribute names must be in lowercase
 - ◆ All tags must be closed (`<br/>`, `<img/>`) while HTML allows `<br>` and `<img>` and implies missing closing tags (`<p>par1 <p>par2`)
 - ◆ XHTML allows only one root `<html>` element (HTML allows more than one)

XHTML vs. HTML (2)

- ◆ Many element attributes are deprecated in XHTML, most are moved to CSS
- ◆ Attribute minimization is forbidden, e.g.

```
<input type="checkbox" checked>
```



```
<input type="checkbox" checked="checked" />
```

- ◆ Note: Web browsers load XHTML faster than HTML and valid code faster than invalid!

The <head> Section

- ◆ Contains information that doesn't show directly on the viewable page
- ◆ Starts after the <!doctype> declaration
- ◆ Begins with <head> and ends with </head>
- ◆ Contains mandatory single <title> tag
- ◆ Can contain some other tags, e.g.
 - ◆ <meta>
 - ◆ <script>
 - ◆ <style>
 - ◆ <!-- comments -->

<head> Section: <title> tag

- ◆ Title should be placed between <head> and </head> tags

```
<title>Title Goes Here</title>
```

- ◆ Used to specify a title in the window title bar
- ◆ Search engines and people rely on titles

<head> Section: <meta>

- ♦ Meta tags additionally describe the content contained within the page

```
<meta name="description" content="HTML  
tutorial" />
```

```
<meta name="keywords" content="html, web  
design, styles" />
```

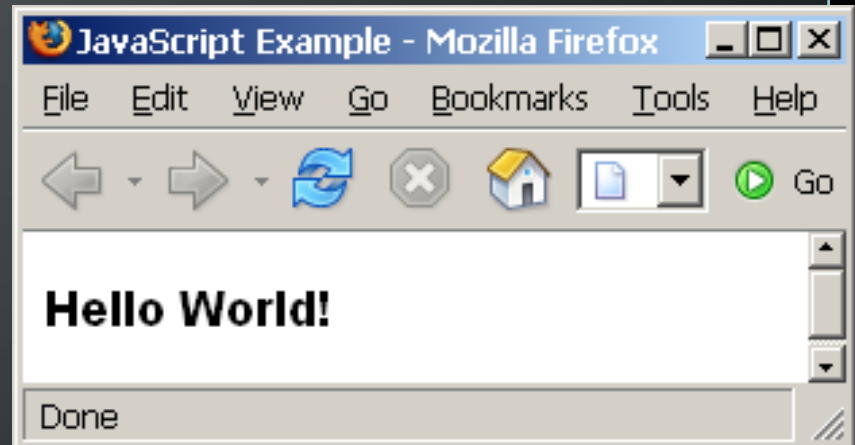
```
<meta name="author" content="Chris Brewer" />
```

<head> Section: <script>

- ◆ The <script> element is used to embed scripts into an HTML document
 - ◆ Script are executed in the client's Web browser
 - ◆ Scripts can live in the <head> and in the <body> sections
- ◆ Supported client-side scripting languages:
 - ◆ JavaScript (it is not Java!)
 - ◆ VBScript
 - ◆ JScript

The <script> Tag – Example

```
<!DOCTYPE HTML>                                     scripts-example.html
<html>
  <head>
    <title>JavaScript Example</title>
    <script type="text/javascript">
      function sayHello() {
        document.write("<p>Hello World!<\n/p>");
      }
    </script>
  </head>
  <body>
    <script type=
      "text/javascript">
      sayHello();
    </script>
  </body>
</html>
```

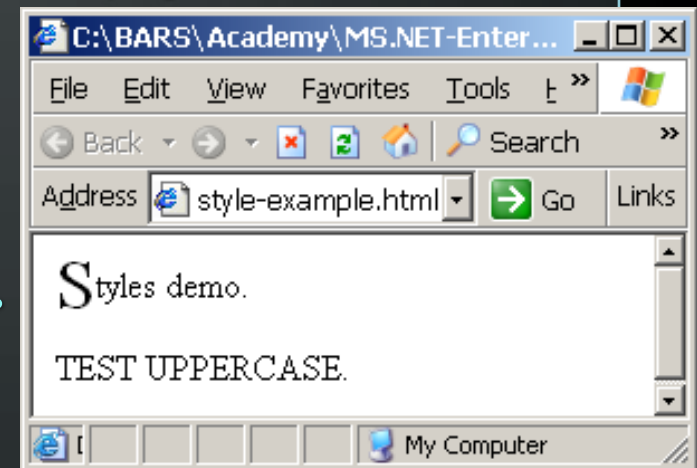


<head> Section: <style>

- ◆ The <style> element embeds formatting information (CSS styles) into an HTML page

```
<html>
  <head>
    <style type="text/css">
      p { font-size: 12pt; line-height: 12pt; }
      p:first-letter { font-size: 200%; }
      span { text-transform: uppercase; }
    </style>
  </head>
  <body>
    <p>Styles demo.<br />
      <span>Test uppercase</span>.
    </p>
  </body>
</html>
```

style-example.html



Comments: `<!-- -->` Tag

- ◆ Comments can exist anywhere between the `<html></html>` tags
- ◆ Comments start with `<!--` and end with `-->`

```
<!-- Telerik Logo (a JPG file) -->  
  
<!-- Hyperlink to the web site -->  
<a href="http://telerik.com/">Telerik</a>  
<!-- Show the news table -->  
<table class="newstable">  
...
```

<body> Section: Introduction

- ◆ The **<body>** section describes the viewable portion of the page
- ◆ Starts after the **<head> </head>** section
- ◆ Begins with **<body>** and ends with **</body>**

```
<html>  
  <head><title>Test page</title></head>  
  <body>  
    <!-- This is the Web page body -->  
  </body>  
</html>
```

Text Formatting

- ◆ Text formatting tags modify the text between the opening tag and the closing tag
 - ◆ Ex. `<b>Hello</b>` makes “Hello” bold

<code></code>	bold
<code><i></i></code>	<i>italicized</i>
<code><u></u></code>	<u>underlined</u>
<code><sup></sup></code>	Sample ^{superscript}
<code><sub></sub></code>	Sample _{subscript}
<code></code>	strong
<code></code>	<i>emphasized</i>
<code><pre></pre></code>	Preformatted text
<code><blockquote></blockquote></code>	Quoted text block
<code></code>	Deleted text – strike-through

Text Formatting – Example

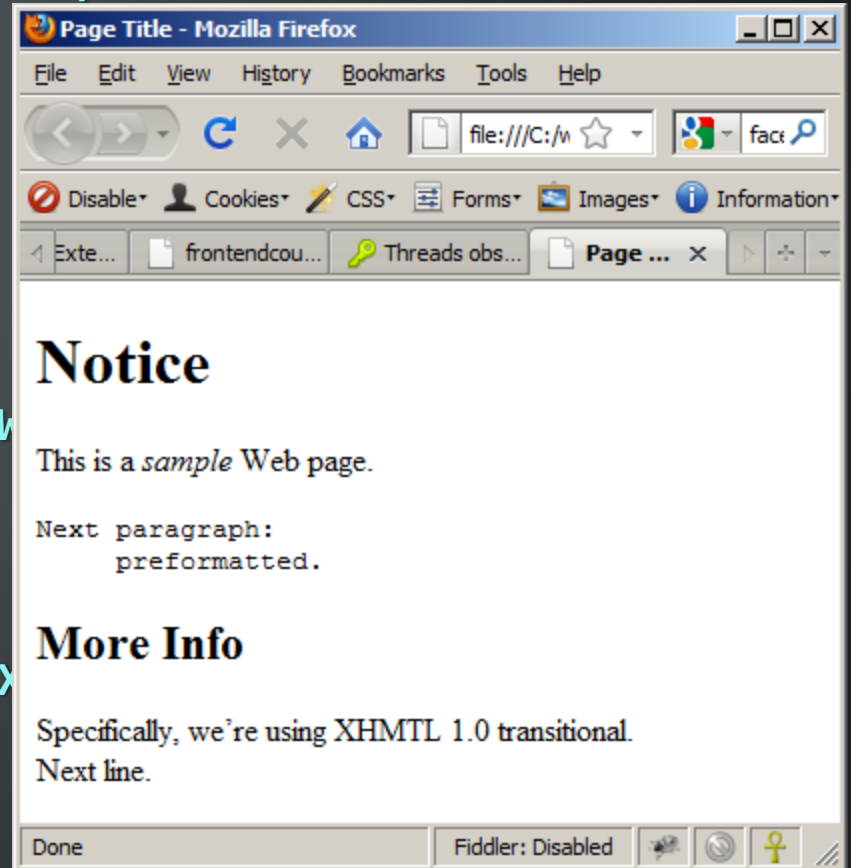
text-formatting.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>Notice</h1>
    <p>This is a <em>sample</em> Web page.</p>
    <p><pre>Next paragraph:
      preformatted.</pre></p>
    <h2>More Info</h2>
    <p>Specifically, we're using XHTML 1.0 transitional.<br />
      Next line.</p>
  </body>
</html>
```


Text Formatting – Example (2)

text-formatting.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>Notice</h1>
    <p>This is a <em>sample</em> Web page.</p>
    <p><pre>Next paragraph:
    preformatted.</pre></p>
    <h2>More Info</h2>
    <p>Specifically, we're using XHTML 1.0 transitional.
    Next line.</p>
  </body>
</html>
```

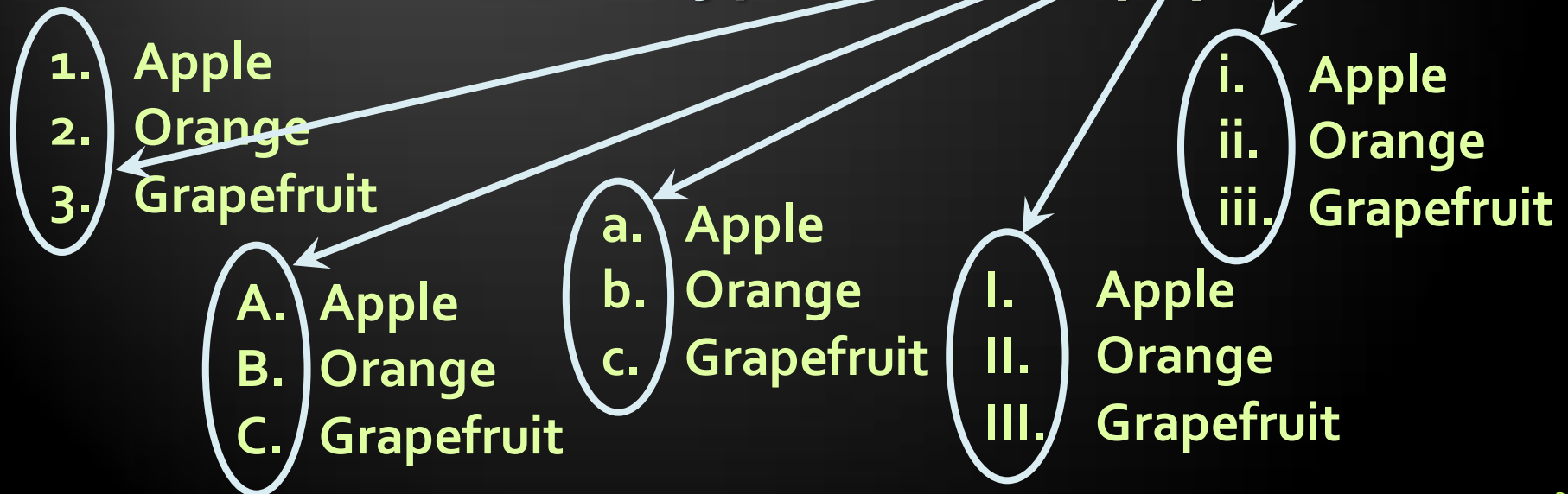


Ordered Lists: Tag

- ◆ Create an Ordered List using :

```
<ol type="1">  
  <li>Apple</li>  
  <li>Orange</li>  
  <li>Grapefruit</li>  
</ol>
```

- ◆ Attribute values for type are 1, A, a, I, or i



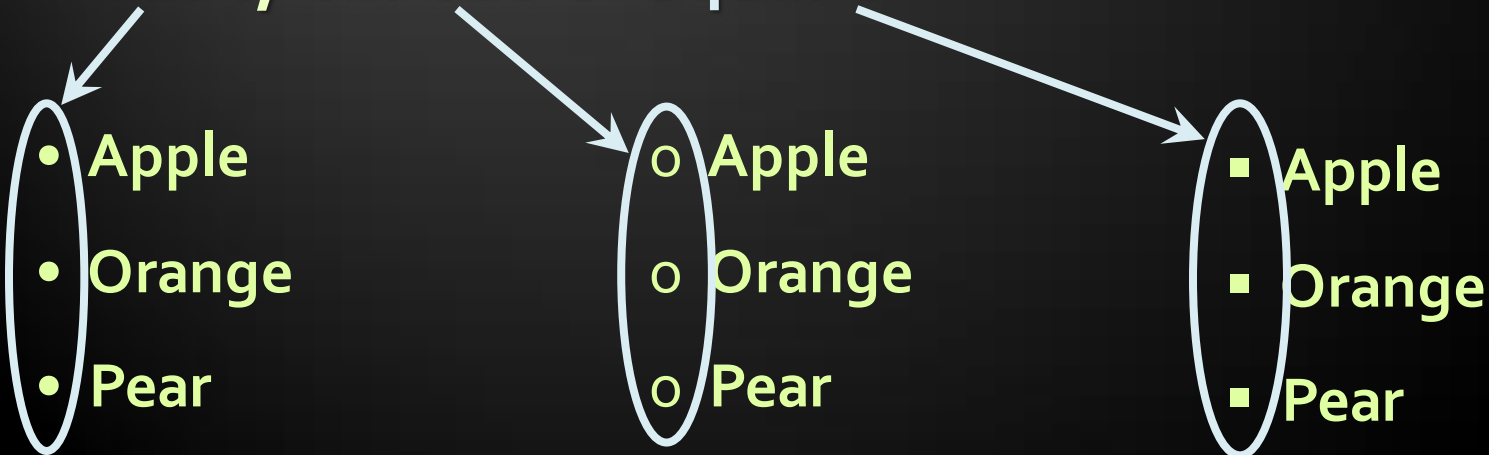
Unordered Lists: Tag

- ◆ Create an Unordered List using :

```
<ul type="disk">  
  <li>Apple</li>  
  <li>Orange</li>  
  <li>Grapefruit</li>  
</ul>
```

- ◆ Attribute values for type are:

- ◆ disc, circle or square



Definition lists: <dl> tag

- ◆ Create definition lists using <dl>
 - ◆ Pairs of text and associated definition; text is in <dt> tag, definition in <dd> tag

```
<dl>  
  <dt>HTML</dt>  
  <dd>A markup language ...</dd>  
  <dt>CSS</dt>  
  <dd>Language used to ...</dd>  
</dl>
```

- ◆ Renders without bullets
- ◆ Definition is indented

Lists – Example

```
<ol type="1">
  <li>Apple</li>
  <li>Orange</li>
  <li>Grapefruit</li>
</ol>

<ul type="disc">
  <li>Apple</li>
  <li>Orange</li>
  <li>Grapefruit</li>
</ul>

<dl>
  <dt>HTML</dt>
  <dd>A markup lang...</dd>
</dl>
```

lists.html

