

Tutorial 1: Developing a Basic Web Page



Computer Science Department
University of Central Florida

COP 3175 – Internet Applications



Objectives

- Create a BASIC one-page HTML document
 - Learn the basic structure of an HTML file
 - Learn what tags are
 - Adding comments to your HTML document
 - Display an HTML file
- Insert an HTML comment
- Work with block-level elements
- Create ordered, unordered, and definition lists



Objectives

- Work with inline elements
- Understand the div and span elements
- Add attributes to HTML elements
- Format page content using the style attribute
- Mark empty elements with one-sided tags
- Add an inline image to a Web page
- Work with character sets and codes



Creating an HTML Document

- Plan out your Web page before you start coding
 - What content do you want on it
 - How do you want it to look
 - Pictures, color scheme, etc.
 - Just like making a layout for a magazine
- Draw a planning sketch or create a sample document using a word processor
- Preparatory work can weed out errors or point to potential problems



Creating an HTML Document

- In planning, identify a document's various elements
 - An **element** is a distinct object in the document, like a paragraph, a heading, or a page's title
 - Even the whole document is considered an element
- The next page shows a flyer for “Dave’s Devil Sticks”
 - We want to convert this flyer to **HTML**
 - So it can be made into a **web page**
 - We start by identifying the various elements on the flyer



Creating an HTML Document

image →

heading → **Who Am I?** **boldface text**

paragraph → Welcome to **Dave's Devil Sticks**. If you are looking for juggling balls, hacky sacks, pins, unicycles, or magic hats, you've come to the wrong place; but if you're looking for high-quality, hand-crafted devil and flower sticks, this is the site for you. I've been designing and building sticks for the past 20 years, and I know that my sticks are the best of their kind. Don't take my word for it; read the following testimonial:

block quote →

I'm more than happy to recommend Dave Vinet's products. I came upon his work 10 years ago and was immediately impressed by his craftsmanship. I've been using his sticks in my shows ever since. They're durable, well-balanced, and attractive props and are the perfect complement to my performances. Thanks Dave!

— Thomas Gage, lead performer at *Circus England*

Every set is checked and tested before being shipped out to assure perfect quality. I take pride in every one of my sticks and I want my customers to feel that same pride.

heading → **My Products**

- bulleted list** →
- **Basic Stick**
The easiest stick to learn with, but "grippy" enough for the most demanding tricks. Comes in red, green, and blue.
 - **Flower Stick**
A graceful stick with colored tassels. Flower Sticks float more slowly, giving you the ideal chance to practice new moves.
 - **Master Stick**
My most popular stick is shorter and heavier for fast play and more advanced tricks. Each Master set is available in custom colors.
 - **Glow Stick**
The Glow Stick shines brightly at night (without the danger of a fire stick). It combines the fun and versatility of the Master Stick, adding the fun tricks unique to a glow-in-the-dark stick.

address → Dave's Devil Sticks • 541 West Highland Dr. • Auburn, ME 04210 • (207) 555 - 9001 •

horizontal line



Marking Elements with Tags

■ HTML Tags

- The core building block of HTML is the **tag**, which marks the presence of an element
 - such as a paragraph, heading, lists, etc
- If the element contains content (such as text), it is marked using a **two-sided tag**
 - A two-sided tag has an “opening tag”
 - indicating the beginning of the content
 - And it has a “closing tag”
 - indicating the end of the content.



Marking Elements with Tags

■ HTML Tags

- Here is the syntax of a two-sided tag:

```
<element>content</element>
```

- where *element* is the name of the element (tag) to be used
- and *content* represents the contents between the tags

■ Example:

- The following code is used to mark a paragraph within an HTML document:

```
<p>Welcome to Dave's Devil Sticks.<\p>
```

- In this example, the <p> tag marks the beginning of the paragraph
- “Welcome to Dave’s Devil Sticks” is the content
- The <\p> tag indicates the end of the paragraph



Marking Elements with Tags

■ HTML Tags

- A two-sided tag should completely enclose its content
- Elements can also contain other elements
- Let's say we wanted to **bold** Dave's business name
 - Here's the code to do so:

```
<p>Welcome to <b>Dave's Devil Sticks<\b>.<\p>
```
 - The web browser now understands to make the display the words "Dave's Devil Sticks" in boldface text.
 - Note:
 - The `` tags have to be completely enclosed (also know as nested) within the `<p>` tags.



Marking Elements with Tags

■ HTML Tags

■ Problem of overlapping tags:

- The `` tags have to be completely enclosed within the `<p>` tags.
- The following syntax would be incorrect:

```
<p>Welcome to <b>Dave's Devil Sticks.<\p><\b>
```
- The closing `<\b>` tag is placed after the closing `<\p>` tag
- This is NOT proper
- Remember:
 - If one tag starts inside (after) another tag,
 - Such as the `` starting inside the `<p>` tag
 - That inside tag must close **FIRST**
 - Before the outer `<\p>` tag closes



The Structure of an HTML File

- All HTML documents must have a **root element**
 - The root element is marked using the `<html>` tag

```
<html>
document content
</html>
```

 - Where *document content* is the content of the entire document.
 - The opening `<html>` tag tells the browser that this is an HTML document
 - The closing `</html>` tag tells a browser when it has reached the end of that HTML document
 - Anything between these two tags makes up the document content, including all other elements, text, and comments



The Structure of an HTML File

- An HTML document is divided into two main sections: the **head** and the **body**
 - The **head element** contains information about the document, for example the document title or the keywords
 - The content of the head element is **not** displayed within the Web page
 - The **body element** contains all of the content to appear on the Web page
 - The body element can contain code that tells the browser how to render (display) the content



The Structure of an HTML File

- An HTML document is divided into two main sections: the **head** and the **body**
 - The overall syntax of an HTML file is as follows:

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



The Structure of an HTML File

- An HTML document is divided into two main sections: the **head** and the **body**
 - The overall syntax of an HTML file is as follows:

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

Note:

- The body element is ALWAYS placed after the head element
- No other elements can be placed between the html, head, and body elements



Defining the Page Title

- An HTML document should have a “title”
 - For this, you use the **title element** as follows:
 - `<title>document title</title>`
 - where document title is the text of the document title
 - A document’s title is usually displayed in the browser’s title bar
 - Reasons for the title:
 - First one is obvious:
 - To give your page a title for customers/friends/etc
 - Also, search engines, like Google and Yahoo!, use the title when displaying the results of your searches



Defining the Page Title

- An HTML document should have a “title”
 - The title tag is part of the **head** content
 - It should go inside the <head> tags as follows

```
<html>
  <head>
    <title>Dave's Devil Sticks</title>
  </head>
  <body>
  </body>
</html>
```

- Note the indentation used to facilitate easy reading of code
- Use three blank spaces
- Or use tabs



Converting an HTML Document into XHTML

- There is considerable overlap between HTML and XHTML
 - Reminder: think of XHTML as a strict version of HTML
 - You can't get away with sloppy (or incorrect) code
 - You must follow rules precisely
 - XHTML files have to be tested against a set of rules that define exactly which markup tags are allowed and how they should work



Converting an HTML Document into XHTML

- There is considerable overlap between HTML and XHTML
 - From HTML to XHTML:
 - You can convert an **HTML** file into an **XHTML** file by simply replacing the opening `<html>` tag with the following three lines of code:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
<html xmlns=http://www.w3.org/1999/xhtml>
```
 - Now, a browser recognizes this as an XHTML doc
 - And strict rules will be applied to make sure your code is correct



Converting an HTML Document into XHTML

- There is considerable overlap between HTML and XHTML
 - From HTML to XHTML:
 - After typing these three lines, a browser now recognizes the this as an XHTML document instead of HTML
 - Beyond these three lines, there is **little difference** between the code of an HTML file and an XHTML file
 - Again, main thing is the strictness requirement of an XHTML document
 - more details coming later...



Adding Comments

- Commenting your code is good practice
 - Comments can include whatever you want:
 - Author's name
 - Document creation date
 - Any other comments (notes that you want)
 - Comments are NOT displayed by the browser
 - Simply used as a tool to annotate code for yourself and others
 - The **comment tag** is used as follows:

`<!-- comment -->`

- where *comment* is the text of the comment or note



Adding Comments

- Commenting your code is good practice

- Example:

- The following comment describes the page we're creating for Dave's business:

```
<-- Page created for Dave Vinet's business -->
```

- Comments can be spread over several lines

```
<-- Dave's Devil Sticks  
a Web Page for Dave Vinet -->
```

- Since comments are ignored by the browser,

- They can be added anywhere in the HTML document



Adding Comments

multiline
comment

```
<html>
<head>
<!-- Dave's Devil sticks
      Author: David Vinet
      Date:   3/1/2011
-->
<title>Dave's Devil sticks</title>
</head>
<body>
</body>
</html>
```



Red text in the code block representing the multiline comment.

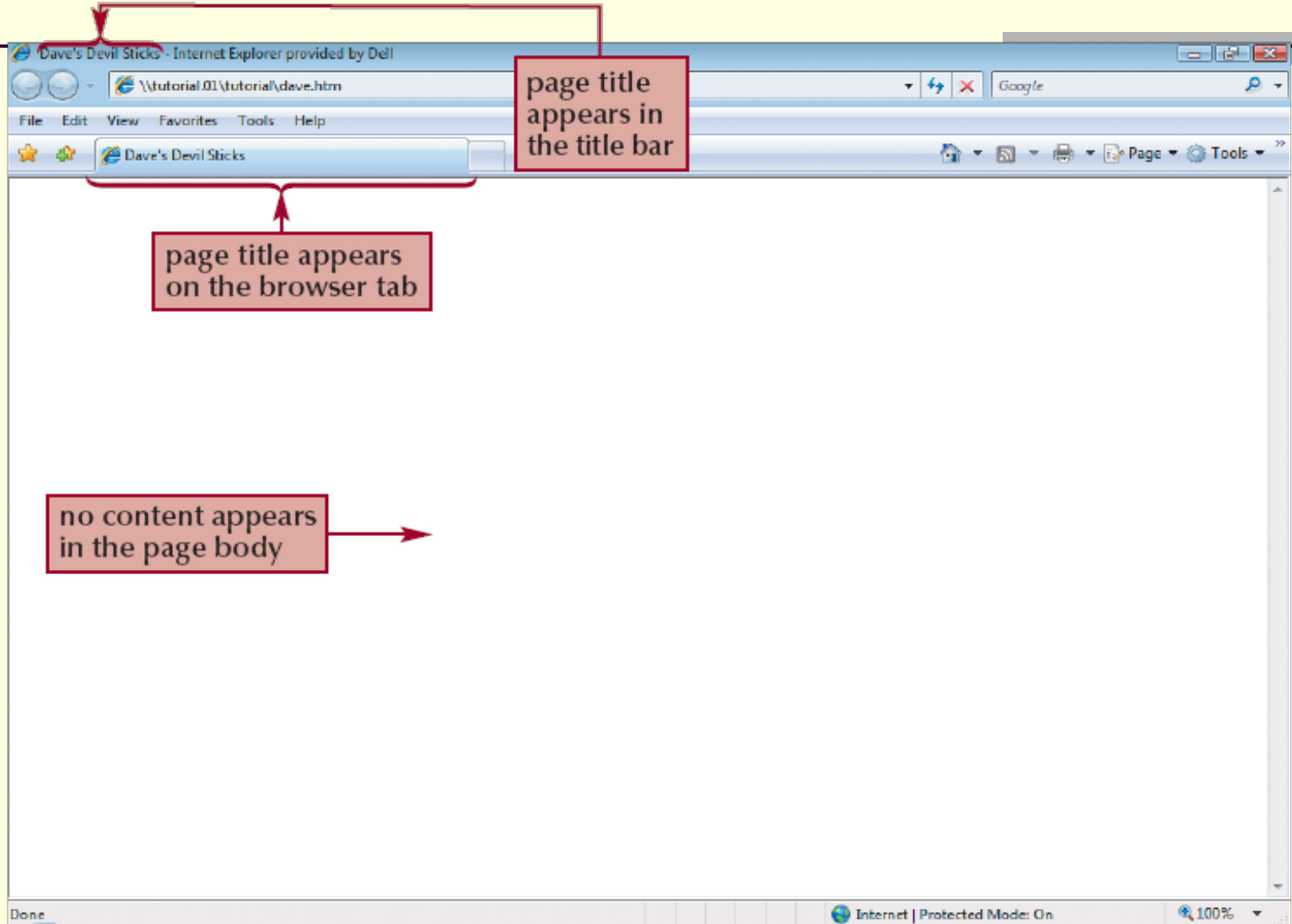


Displaying an HTML File

- As you continue modifying the HTML code, you should occasionally view it with your Web browser to verify that you have not introduced any errors
 - You may want to view the results using different browsers to check for compatibility
 - From Notepad++
 - Go to Run and click Launch in IE
 - Also Run it in Chrome and Firefox



Displaying an HTML File





Adding Content to HTML file

- With the basic structure of the HTML file now in place, we're ready to add content
- For now, we'll look at the following elements:
 - 1) Block-Level elements
 - 2) Lists
 - 3) Inline elements



Working with Block-Level Elements

- **Block-level** elements are elements that contain content that is viewed as a distinct block within the Web page
 - When rendered visually, block-level elements always start on a new line, within the document
 - Examples of block level elements:
 - Paragraphs
 - headings

Note:

- Check out the demo_html.htm file from the tutorial.01/demo folder
- You can type in heading elements and see the results



Working with Headings

- **Heading elements** are block-level elements that contain the text of main headings on the Web page
 - Think of headings as **titles** in a typical document
 - They are often used for introducing new topics
 - The syntax for a head is as follows:
 - `<h n >content<h n >`
 - where n is an integer between 1 and 6
 - `<h1>` is the largest heading
 - Usually in big bold text
 - `<h2>` down to `<h6>` get progressively smaller
 - Used for subheadings within the document



Marking Block-Level Elements

- To mark a heading, enter
`<hn>content</hn>`
 - where *n* is an integer from 1 to 6 and *content* is the text of heading
- To mark a paragraph, enter
`<p>content</p>`
- To mark a block quote, enter
`<blockquote>content</blockquote>`
- To mark a generic block-level element, enter
`<div>content</div>`



Marking-Up Dave's Page

- We start by adding the headings to Dave's page:
 - He has three headings he needs:
 - The first is an h1 heading
 - Contains the company name
 - The other two are h2 headings
 - For different subsections of his page
- Next we add the paragraph elements:
 - Note:
 - In old HTML code, you may see an opening `<p>` tag without a closing `<\p>` tag
 - This does violate HTML rules; you must use closing tag.



Adding `<h1>` and `<h2>` Markup Tags

```
<body>  
  <h1>Dave's Devil sticks</h1>  
  <h2>Who Am I?</h2>  
  <h2>My Products</h2>  
</body>
```

Dave's Devil Sticks ← h1 heading

Who Am I?
My Products } h2 headings



Marking Paragraph Elements

paragraphs

```
<body>
  <h1>dave's devil sticks</h1>
  <h2>who Am I?</h2>
  <p>welcome to Dave's Devil sticks. If you are looking for juggling balls,
    hacky sacks, pins, unicycles, or magic hats, you've come to the wrong
    place; but if you're looking for high-quality, hand-crafted devil and
    flower sticks, this is the site for you. I've been designing and building
    sticks for the past 20 years, and I know that my sticks are the best of
    their kind.</p>
  <p>Every set is checked and tested before being shipped out to assure perfect
    quality. I take pride in every one of my sticks and I want my customers to
    feel that same pride.</p>
  <h2>My Products</h2>
</body>
```

Dave's Devil Sticks

Who Am I?

Welcome to Dave's Devil Sticks. If you are looking for juggling balls, hacky sacks, pins, unicycles, or magic hats, you've come to the wrong place; but if you're looking for high-quality, hand-crafted devil and flower sticks, this is the site for you. I've been designing and building sticks for the past 20 years, and I know that my sticks are the best of their kind.

Every set is checked and tested before being shipped out to assure perfect quality. I take pride in every one of my sticks and I want my customers to feel that same pride.

My Products

paragraphs



White Space and HTML

- HTML file documents are composed of text characters and **white space**
 - **White space** is the blank space, tabs, and line breaks within the file
 - HTML treats each occurrence of **white space** as a single blank space
 - You can use **white space** to make your document more readable
 - Just like when making a flyer or a magazine layout
 - White space is actually essential to make the flyer, document, etc. more readable



White Space and HTML

■ Example:

HTML Code	Preview
<pre data-bbox="311 529 962 882"><p>CGS 3175 is AWESOME</p> <p>CGS 3175 is AWESOME</p> <p>CGS 3175 is AWESOME</p></pre> <p data-bbox="320 919 539 953">Preview Code</p>	<p data-bbox="1058 551 1354 575">CGS 3175 is AWESOME</p> <p data-bbox="1058 619 1354 644">CGS 3175 is AWESOME</p> <p data-bbox="1058 688 1354 712">CGS 3175 is AWESOME</p>

- The point is to see that these three “paragraphs” are **all rendered the same** by a Web browser
 - So you can use white space however you wish
 - Without it impacting the appearance of a Web page



Marking a Block Quote

image →

heading → **Who Am I?** **boldface text**

paragraph → Welcome to **Dave's Devil Sticks**. If you are looking for juggling balls, hacky sacks, pins, unicycles, or magic hats, you've come to the wrong place; but if you're looking for high-quality, hand-crafted devil and flower sticks, this is the site for you. I've been designing and building sticks for the past 20 years, and I know that my sticks are the best of their kind. Don't take my word for it; read the following testimonial:

block quote →

I'm more than happy to recommend Dave Vinet's products. I came upon his work 10 years ago and was immediately impressed by his craftsmanship. I've been using his sticks in my shows ever since. They're durable, well-balanced, and attractive props and are the perfect complement to my performances. Thanks Dave!

— Thomas Gage, lead performer at *Circus England*

horizontal line

horizontal line

horizontal line

address →

Dave's Devil Sticks • 541 West Highland Dr. • Auburn, ME 04210 • (207) 555 - 9001 •



Marking a Block Quote

- Dave wants to display the customer quote
 - This needs to be indented
 - What did you just learn from the white space slide?
 - You can't use spaces...they get **ignored** by the browser
 - So you must use an HTML tag to achieve the effect
 - Again, the syntax for a quote is as follows:
 - `<blockquote>content</blockquote>`
 - Note:
 - Some browsers display quotes differently
 - The only way to guarantee the consistency of your quotes (and other content) is to **use styles**.....coming later



Marking a Block Quote

block quote

```
<p>welcome to Dave's Devil Sticks. If you are looking for juggling balls,
hacky sacks, pins, unicycles, or magic hats, you've come to the wrong
place; but if you're looking for high-quality, hand-crafted devil and
flower sticks, this is the site for you. I've been designing and building
sticks for the past 20 years, and I know that my sticks are the best of
their kind. Don't take my word for it; read the following testimonial:</p>
<blockquote>
<p>I'm more than happy to recommend Dave Vinet's products. I came upon his
work 10 years ago and was immediately impressed by his craftsmanship.
I've been using his sticks in my shows ever since. They're durable,
well-balanced, and attractive props and are the perfect complement to
my performances. Thanks Dave!</p>
<p>Thomas Gage, lead performer at Circus England</p>
</blockquote>
<p>Every set is checked and tested before being shipped out to assure perfect
quality. I take pride in every one of my sticks and I want my customers to
feel that same pride.</p>
```

Dave's Devil Sticks

Who Am I?

Welcome to Dave's Devil Sticks. If you are looking for juggling balls, hacky sacks, pins, unicycles, or magic hats, you've come to the wrong place; but if you're looking for high-quality, hand-crafted devil and flower sticks, this is the site for you. I've been designing and building sticks for the past 20 years, and I know that my sticks are the best of their kind. Don't take my word for it; read the following testimonial:

I'm more than happy to recommend Dave Vinet's products. I came upon his work 10 years ago and was immediately impressed by his craftsmanship. I've been using his sticks in my shows ever since. They're durable, well-balanced, and attractive props and are the perfect complement to my performances. Thanks Dave!

Thomas Gage, lead performer at Circus England

Every set is checked and tested before being shipped out to assure perfect quality. I take pride in every one of my sticks and I want my customers to feel that same pride.

block quote



Marking-Up a List

- HTML supports three kinds of lists: **ordered**, **unordered**, and **definition**
 - You use an **ordered list** for items that must appear in a numerical order
 - You use an **unordered list** for items that do not need to occur in any special order
 - One **list** can contain another list
 - This is called a nested list



Creating a Definition List

- The **definition list** contains a list of terms, each followed by the term's description
- Web browsers typically display the definition description below the definition term and slightly indented:
- Example:
 - Basic Stick**
 - Easiest stick to learn**



Marking a List

Marking Lists

- To mark an ordered list, enter

```
<ol>
  <li>item1</li>
  <li>item2</li>
  ...
</ol>
```

where *item1*, *item2*, and so forth are the items in the list.

- To mark an unordered list, use

```
<ul>
  <li>item1</li>
  <li>item2</li>
  ...
</ul>
```

- To mark a definition list, use

```
<dl>
  <dt>term1</dt>
  <dd>description1</dd>
  <dt>term2</dt>
  <dd>description2a</dd>
  <dd>description2b</dd>
  ...
</dl>
```

where *term1*, *term2*, etc. are the terms in the list and *description1*, *description2a*, *description2b*, etc. are the descriptions associated with each term.



Marking-Up a List

■ Ordered List Example:

H T M L C o d e

```
<p>Why is this class AWESOME:</p>
<ol>
  <li>The instructor...duh!</li>
  <li>We learn how to make websites!</li>
  <li>The free cookies</li>
  <li>Did we mention the instructor?</li>
</ol>
```

Preview Code

P r e v i e w

Why is this class AWESOME:

1. The instructor...duh!
2. We learn how to make websites!
3. The free cookies
4. Did we mention the instructor?



Marking-Up a List

■ Unordered List Example:

H T M L C o d e

```
<p>Why is this class AWESOME:</p>
<ul>
  <li>The instructor...duh!</li>
  <li>We learn how to make websites!</li>
  <li>The free cookies</li>
  <li>Did we mention the instructor?</li>
</ul>
```

Preview Code

P r e v i e w

Why is this class AWESOME:

- The instructor...duh!
- We learn how to make websites!
- The free cookies
- Did we mention the instructor?



Marking-Up a List

■ Nested List Example:

```
HTML Code
```

```
<p>Why is this class AWESOME:</p>
<ul>
  <li>The instructor...duh!</li>
  <li>We learn how to make websites!</li>
  <li>The free cookies</li>
  <ul>
    <li>Chocolate Chip</li>
    <li>Oatmeal</li>
    <li>Ginger Snaps</li>
  </ul>
  <li>Did we mention the instructor?</li>
</ul>
```

Preview Code

```
Preview
```

Why is this class AWESOME:

- The instructor...duh!
- We learn how to make websites!
- The free cookies
 - Chocolate Chip
 - Oatmeal
 - Ginger Snaps
- Did we mention the instructor?



Marking-Up a List

■ Definition List Example:

H T M L C o d e

```
<dl>
  <dt>Basic Stick</dt>
  <dd>Easiest stick to learn</dd>
  <dt>Flower Stick</dt>
  <dd>A graceful stick with tassels</dd>
  <dt>Master Stick</dt>
  <dd>Our most popular stick</dd>
</dl>
```

Preview Code

P r e v i e w

Basic Stick
 Easiest stick to learn

Flower Stick
 A graceful stick with tassels

Master Stick
 Our most popular stick



Using Other Block-Level Elements

- HTML supports the **address element** to indicate contact information
 - Most browsers display an address element in an italicized font, and some right-justify or indent addresses
 - Note that the address just runs together for now
 - We'll make it more readable later

```
<address>Dave's Devil sticks  
541 West Highland Dr.  
Auburn, ME 04210  
(207) 555 - 9001  
</address>  
</body>  
</html>
```

My Products

- Basic Stick
- Flower Stick
- Master Stick
- Glow Stick

address text is displayed
in an italics font

Dave's Devil Sticks 541 West Highland Dr. Auburn, ME 04210 (207) 555 - 9001



Using Other Block-Level Elements

Block-Level Element	Marks	Usual Visual Appearance
<code><address> ... </address></code>	Contact information	<i>Italicized text</i>
<code><blockquote> ... </blockquote></code>	An extended quotation	Plain text indented from the left and right
<code><center> ... </center></code>	Text horizontally centered with the block (deprecated)	Plain text, centered
<code><dd> ... </dd></code>	A definition description	Plain text
<code><dir> ... </dir></code>	A multicolumn directory list (deprecated)	Plain text
<code><div> ... </div></code>	A generic block-level element	Plain text
<code><dl> ... </dl></code>	A definition list	Plain text
<code><dt> ... </dt></code>	A definition term from a definition list	Plain text
<code><h1> ... </h1></code>	A heading where <i>n</i> is a value from 1 to 6 with h1 as the most prominent heading and h6 the least prominent	Boldfaced text of various font sizes
<code> ... </code>	A list item from an ordered or unordered list	Bulleted or numbered text
<code><menu> ... </menu></code>	A single column menu list (deprecated)	Plain text
<code> ... </code>	An ordered list	Plain text
<code><p> ... </p></code>	A paragraph	Plain text
<code><pre> ... </pre></code>	Preformatted text, retaining all white space and special characters	Fixed width text
<code> ... </code>	An unordered list	Plain text



Working with Inline Elements

- **Block-level** elements place content on a new line within the page
- Another type of element is an **inline element**
 - Marks a section of text within a block-level element
 - Think of block-level elements as paragraphs
 - And think of inline elements as text within said paragraphs
 - Inline elements do NOT start on a new line
 - Rather, they flow “in-line” with the rest of the text in the block



Working with Inline Elements

- An **inline element** marks a section of text within a block-level element
 - Often used to format characters and words
 - Also referred to as **character formatting elements**

The image shows a side-by-side comparison of HTML code and its rendered output. On the left, a window titled "HTML Code" contains the following code: `<p>Welcome to Dave's Devil Sticks, owned and operated by <i>David Vinet</i>.</p>`. Red annotations include a box pointing to `` with the text "text marked with the tag" and another box pointing to `<i>` with the text "text marked with the <i> tag". A "Preview Code" button is at the bottom. On the right, a window titled "Preview" shows the rendered text: "Welcome to **Dave's Devil Sticks**, owned and operated by *David Vinet*." Red annotations include a box pointing to "Dave's Devil Sticks" with the text "bold text" and another box pointing to "David Vinet" with the text "italicized text".



Working with Inline Elements

Inline Element	Marks	Usual Visual Appearance
<abbr> ... </abbr>	An abbreviation	Plain text
<acronym> .. </acronym>	An acronym	Plain text
 ... 	Boldfaced text	Boldfaced text
<big> ... </big>	Big text	Larger text
<cite> ... </cite>	A citation	<i>Italicized text</i>
<code> ... </code>	Program code	Fixed width text
 ... 	Deleted text	Strikethrough text
<dfn> ... </dfn>	A definition term	<i>Italicized text</i>
 ... 	Emphasized content	<i>Italicized text</i>
<i> ... </i>	Italicized text	<i>Italicized text</i>
<ins> ... </ins>	Inserted text	<u>Underlined text</u>
<kbd> ... </kbd>	Keyboard-style text	Fixed width text
<q> ... </q>	Quoted text	"Quoted text"
<s> ... </s>	Strikethrough text (Deprecated)	Strikethrough text
<samp> ... </samp>	Sample computer code	Fixed width text
<small> ... </small>	Small text	Smaller text
 ... 	A generic inline element	Plain text
<strike> ... </strike>	Strikethrough text (Deprecated)	Strikethrough text
 ... 	Strongly emphasized content	Boldfaced text
_{...}	Subscripted text	<small>Subscripted text</small>
^{...}	Superscripted text	<small>Superscripted text</small>
<tt> ... </tt>	Teletype text	Fixed width text
<u> ... </u>	Underlined text (Deprecated)	<u>Underlined text</u>
<var> ... </var>	Programming variables	<i>Italicized text</i>



Using Generic Elements: div & span

- Div and Span elements
 - Developers often want a generic element to control a block or a string of inline text
 - **div** is used to make a generic block-level element
`<div>content</div>`
 - **span** is used to make a generic inline element
`content`
 - These tags do not assign any default content
 - What is the purpose?
 - Developers can utilize styles, within these elements, to fully control the appearance of the Web page
 - Again, styles are coming in a bit...



Logical Elements vs Physical Elements

- Logical elements
 - describes the nature of the enclosed content, but not necessarily how that content should appear
- Physical elements
 - describes how content should appear, but doesn't indicate the content's nature
- Which to use?
 - You should use a logical element that accurately describes the enclosed content whenever possible, and use physical elements only for general content



Using Element Attributes

- Many tags contain attributes
 - control the use, behavior, and in some cases the appearance, of elements in the document
- Attributes are inserted within the tag brackets

`<element attribute1="value1" attribute2="value2" ...>content</element>`

- Where attribute1 and attribute2 are the names of the attributes associated with the given element
- And value1, value2, etc are the values of those attributes
- Attributes can be listed in any order
- But they must be separated by spaces



Using Element Attributes

- ID attribute

- Very common attribute

- Uniquely identifies an element within a Web page

- Look at the following h1 tag:

- ```
<h1 id="mainhead">Dave's Devil Sticks</h1>
```

- This code assigns the id value of “mainhead” to the h1 heading “Dave’s Devil Sticks”

- This distinguishes this heading from other headings on the page

- In the next chapter, we’ll learn benefits of this attribute



# The Style Attribute

---

- Arguably the most important attribute
  - Use the **style attribute** to control the appearance of an element, such as text alignment

- Syntax of style attribute:

```
<element style="rules" ...>content</element>
```

- where **rules** is a set of style rules that you choose

- Style rules:

- Style rules are entered by stating the style name
- followed by a colon and then a style value
- You can have multiple style rules with each name/value pair separated by a semicolon

- Here's the general syntax:

```
style="name1:value1; name2:value2; ..."
```



# The Style Attribute

---

- Example Styles:

- There are MANY styles that you will learn throughout the semester
  - For now, we focus on only a small set of them
- The **text-align style** tells the browser how to horizontally align the contents of an element

- Syntax:

`style="text-align: alignment"`

- where *alignment* is left, right, center or justify

- Example using text-align on an h1 tag:

```
<h1 style="text-align: center">content</h1>
```



# The Style Attribute

---

- Example Styles:
  - The **color style** tells the browser to render the text in a certain color
  - Syntax:
    - style="color: color"
    - where color is a color name, such as red, blue, green, etc
    - Here's how you would make an h3 tag render as red  
`<h3 style="color: red">content</h3>`



# The Style Attribute

---

- Example Styles:

- Combining multiple styles

- As mentioned you can combine multiple style rules within one style declaration
    - Here's how you would make an h2 tag render as blue, while aligning it to the center

```
<h2 style="text-align: center; color: red">...</h2>
```

- Old method:

- Early versions of HTML would use different methods of changing the appearance of content
    - These are all depreciated and will not be covered
    - You need to use styles!



# The Style Attribute

The screenshot shows a web development tool with two main panels: "HTML Code" and "Preview".

**HTML Code Panel:** The title bar reads "HTML Code". The code editor contains the following HTML snippet: `<h1 style="text-align: center; color: red">Dave's Devil Sticks</h1>`. A red box highlights the style attribute, with an arrow pointing to it from a text box that says "styles to center the text and change the font color to red". Below the code editor is a "Preview Code" button.

**Preview Panel:** The title bar reads "Preview". It displays the rendered output of the code: the text "Dave's Devil Sticks" in a red, serif font, centered on the page. A red bracket is drawn under the text, and an arrow points from a red box labeled "formatted text" to the center of the bracket.



# Working with Empty Elements

---

- An **empty element** contains no content
  - Empty elements appear in code as **one-sided tags**
- There are several useful empty elements:
  - Line Break:
    - The one-sided tag to mark a line break is  
`<br />`
    - This is useful to go to the next line without leaving the current “block”
      - Meaning, using the `<br />` tag allows you to go to the next immediate line...without adding an extra space between lines
    - Line breaks should go within block-level elements only
      - Such as paragraphs or headings



# Working with Empty Elements

---

- An **empty element** contains no content
  - Empty elements appear in code as **one-sided tags**
- There are several useful empty elements:
  - Horizontal Rule element:
    - The horizontal rule element places a horizontal line across the Web page  
`<hr />`
    - Useful in breaking up information on pages into topical sections



# Working with Empty Elements

```
<blockquote>
 <hr />
 <p>I'm more than happy to recommend Dave Vinet's products. I came upon his
 work 10 years ago and was immediately impressed by his craftsmanship.
 I've been using his sticks in my shows ever since. They're durable,
 well-balanced, and attractive props and are the perfect complement to
 my performances. Thanks Dave!</p>

 <p>Thomas Gage, lead performer at <i>Circus England</i></p>
 <hr />
</blockquote>

<p>Every set is checked and tested before being shipped out to assure perfect
quality. I take pride in every one of my sticks and I want my customers to
feel that same pride.</p>

<h2>My Products</h2>

 Basic Stick

 The easiest stick to learn with, but "grippy" enough for the most demanding
 tricks. Comes in red, green, and blue.
 Flower Stick

 A graceful stick with colored tassels. Flower sticks float more slowly, giving
 you the ideal chance to practice new moves.
 Master Stick

 My most popular stick is shorter and heavier for fast play and more advanced
 tricks. Each Master set is available in custom colors.
 Glow Stick

 The Glow Stick shines brightly at night (without the danger of a fire stick).
 It combines the fun and versatility of the Master Stick, adding the fun tricks
 unique to a glow-in-the-dark stick.

<hr />

<address style="text-align: center;">Dave's Devil Sticks
 541 West Highland Dr.
 Auburn, ME 04210
 (207) 555 - 9001
</address>
</body>
```



# Working with Empty Elements

---

- Inserting **Images** into a document:
  - HTML files are simple text files
    - Non-textual elements, such as graphics, must be stored in separate files and loaded by the Web browser
  - Common image formats are:
    - GIF or JPEG (JPG)
    - You can use an image editing application (outside the scope of this class) to modify your images and size the appropriately.
      - Example programs are the famous (and expensive) Adobe Photoshop
      - Free programs include the opensource GIMP
    - Images should be a resolution of 72 dpi



# Working with Empty Elements

---

- Inserting Images into a document:
  - To display a graphic, you insert an **inline image** into the page
  - An **inline image** displays a graphic image, located in a separate file, within the page

```

```

    - where *file* is the name of the image file
    - and *text* is the text displayed, in lieu of the graphic image
    - For now, we assume that the graphic file is located in the SAME folder as the html document



# Working with Empty Elements

center the contents  
of the h1 element

inline image  
element

```
<body>
<h1 style="text-align: center"></h1>
<h2>who Am I?</h2>
<p>welcome to Dave's Devil sticks. If you are looking for juggling balls,
hacky sacks, pins, unicycles, or magic hats, you've come to the wrong
place; but if you're looking for high-quality, hand-crafted devil and
flower sticks, this is the site for you. I've been designing and building
sticks for the past 20 years, and I know that my sticks are the best of
their kind. Don't take my word for it; read the following testimonial:</p>
```



logo centered  
on page



## Who Am I?

Welcome to **Dave's Devil Sticks**. If you are looking for juggling balls, hacky sacks, pins, unicycles, or magic hats, you've come to the wrong place; but if you're looking for high-quality, hand-crafted devil and flower sticks, this is the site for you. I've been designing and building sticks for the past 20 years, and I know that my sticks are the best of their kind. Don't take my word for it; read the following testimonial:



# Working with Character Sets and Special Characters

---

## ■ Character sets:

- Thus far we've only entered basic text, numbers and known punctuation
- But there are many other characters (symbols) that need to be represented
  - ©, ®, ¢, ∞, ±, ≤, ≥, and the list goes on
- So how do you “type” these characters?
  - You use **character sets**
  - come in a wide variety of sizes, based on the number of symbols required for communication in the chosen Language



# Working with Character Sets and Special Characters

---

## ■ Character sets:

### ■ Example character sets:

- ASCII (American Standard Code for Information Interchange)
  - Represents the alphabet of the English language
- Latin-1 or ISO 8859-1
  - More extended character sets
  - Support 255 characters and can be used by most languages that employ a Latin alphabet
- Unicode
  - Most extensive character set...supports 65,536 symbols
- UTF-8
  - Compressed version of Unicode and most common on Web



# Working with Character Sets and Special Characters

---

- Character sets:

- To store a character set, browsers need to associate each symbol with a number in a process called **character encoding**

- For example, the copyright symbol from the UTF-8 character set has the number 169.
- You can insert this symbol by typing the following in your HTML file:

**&#169;**

- This would make the © symbol appear on the browser



# Working with Character Sets and Special Characters

---

## ■ Character sets:

### ■ Another way to insert a special symbol is to use a **character entity reference**

- This is a short memorable name, which is used in place of the numeric character reference

- Example for the copyright symbol:

**&copy;**

- Again, this inserts a © symbol into your HTML document

- Advantages of character entity references:

- Browsers can know them without knowing the character set
- Easy to remember

- Disadvantage is that old browsers may not recognize it

- Instead of a © appearing, you would actually see &copy; in the doc



# Working with Character Sets and Special Characters

---

## ■ Special characters

- Let's say you wanted to make a website that teaches html
  - And you were explaining the use of the `<h1>` tag
  - You cannot simply type:  
"The `<h1>` tag is used to mark h1 headings."
    - The browser will understand the `<h1>` as the beginning of an actual `<h1>` tag and will make the rest of the sentence a heading!
  - Instead, you have to use codes for the `<` and `>` symbols:
  - The text would be:  
"The `&lt;h1&gt;` tag is used to mark h1 headings."
  - This would then appear properly in your browser



# Working with Character Sets and Special Characters

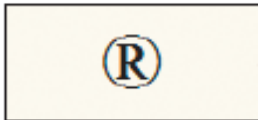
## Inserting Character Codes

- To insert a character based on a numeric character reference, use `&#code;` where *code* is the character code number.
- To insert a character based on the character entity reference, use `&char;` where *char* is the name assigned to the character.
- To insert a nonbreaking space, use `&nbsp;`
- To insert the < symbol, use `&lt;`
- To insert the > symbol, use `&gt;`



# Working with Character Sets and Special Characters

Enter a character code or character name:



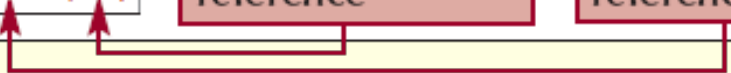
character symbol

character entity reference

Select a table of characters from the list box:

## General Symbols

" quot (34)	& amp (38)	< lt (60)	> gt (62)	nbsp (160)	¡ iexcl (161)	¢ cent (162)	£ pound (163)	¤ curren (164)	¥ yen (165)
 brvbar (166)	§ sect (167)	¨ uml (168)	© copy (169)	ª ordf (170)	« laquo (171)	¬ not (172)	shy (173)	® reg (174)	¯ macr (175)
° deg (176)	± plusmn (177)	² <sup>2</sup> sup2 (178)	³ <sup>3</sup> sup3 (179)	´ acute (180)	µ <sup>µ</sup> micro (181)	¶ para (182)	· middot (183)	¸ <sup>¸</sup> cedil (184)	¹ <sup>1</sup> sup1 (185)
º <sup>º</sup> ordm (186)	» raquo (187)	¼ <sup>1/4</sup> frac14 (188)	½ <sup>1/2</sup> frac12 (189)	¾ <sup>3/4</sup> frac34 (190)	numeric character reference		character entity reference		





# Tutorial Summary

---

- Create a basic Web page using HTML
- Concepts and history surrounding networks and the development of the World Wide Web
- History of HTML
- Creation of a simple Web page
- Block-level elements
- Inline elements
- Element attributes
- Character sets and special character symbols



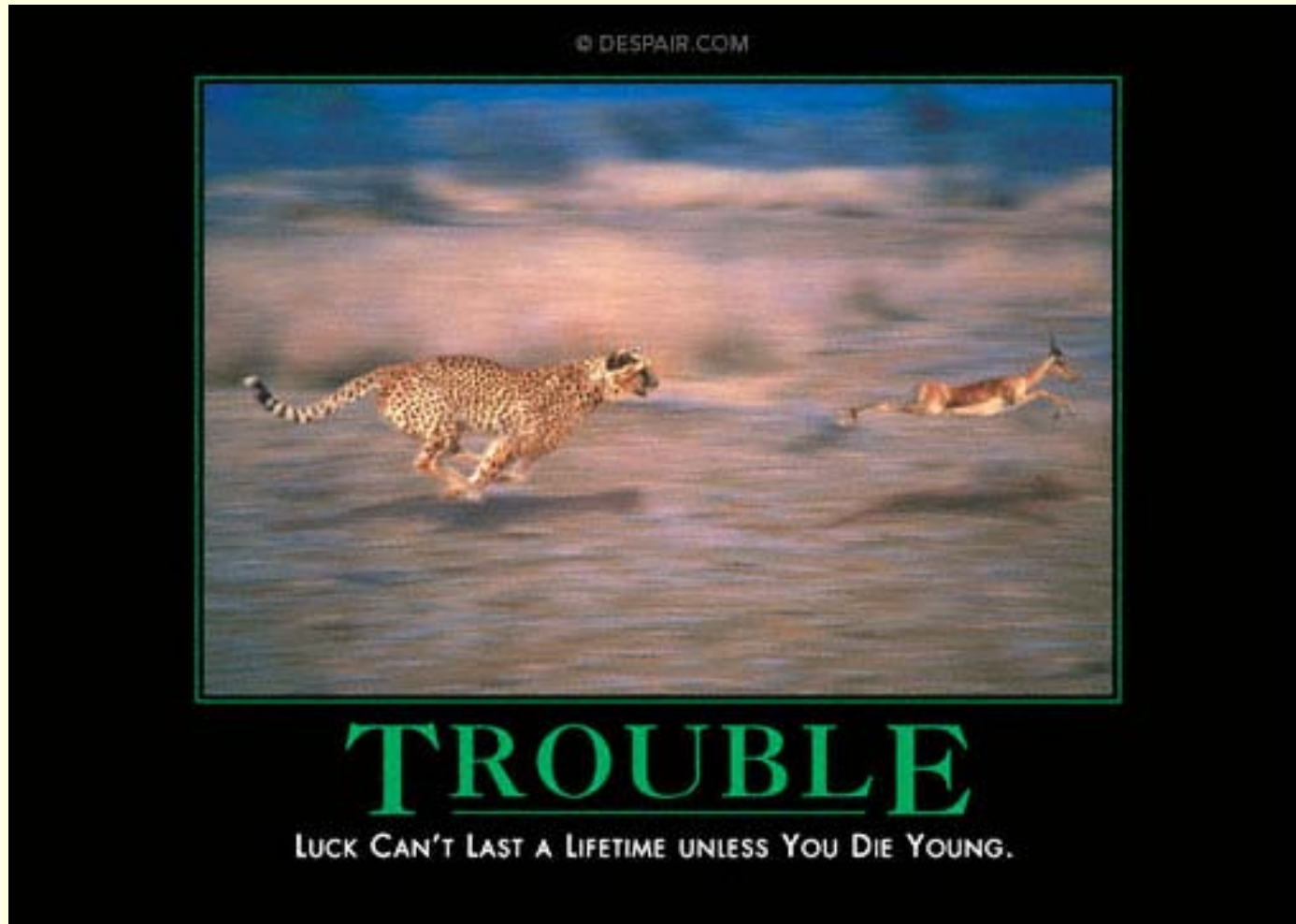
# Developing a Basic Web Page

---

**WASN'T  
THAT  
GREAT!**



# Daily Demotivator



# Tutorial 1: Developing a Basic Web Page



Computer Science Department  
University of Central Florida

*COP 3175 – Internet Applications*