Introduction to HTML and Markup

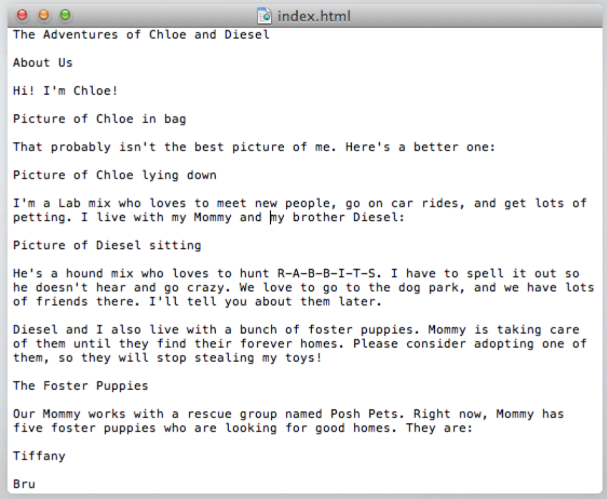
CSE/ISE 102: Introduction to Web Design and Programming (Section 02)
Stony Brook University

Building a Page, Step by Step

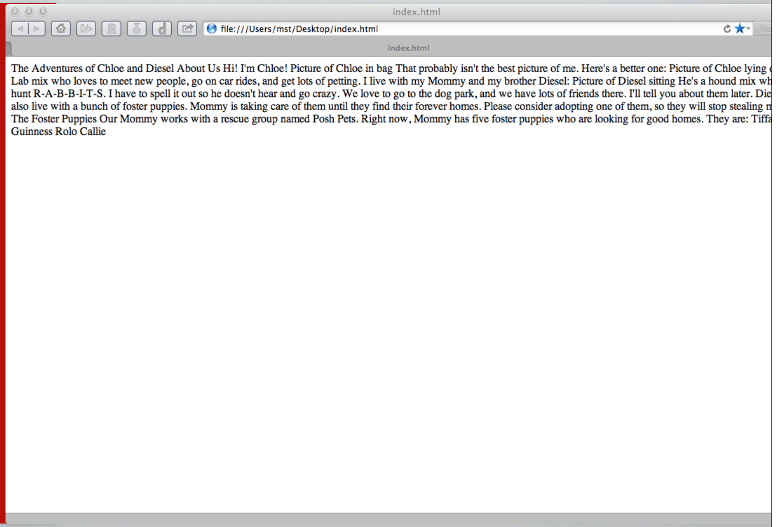
- We will follow five basic steps for our example:
  - Start with content
  - Give the document structure
  - Identify text elements
  - Add an image
  - Change the page appearance using a style sheet
- Start by launching a text editor (e.g., Notepad or TextEdit)

Step 1: Start with Content

- In our text editor, we'll type up the basic content for our site
  - if you're using TextEdit on the Mac, be sure to select "Make Plain Text" from the "Format" menu
- When we're done, we'll fire up a Web browser and see how our new page looks...
- Important: Be sure to save your file with a ".htm" or ".html" extension
  - filenames are usually case-sensitive
  - filenames MUST not contain spaces or special characters



```
The Adventures of Chloe and Diesel

About Us

Hi! I'm Chloe!

Picture of Chloe in bag

That probably isn't the best picture of me. Here's a better one:

Picture of Chloe lying down

I'm a Lab mix who loves to meet new people, go on car rides, and get lots of
petting. I live with my Mommy and my brother Diesel:

Picture of Diesel sitting

He's a hound mix who loves to hunt R-A-B-B-I-T-S. I have to spell it out so
he doesn't hear and go crazy. We love to go to the dog park, and we have lots
of friends there. I'll tell you about them later.

Diesel and I also live with a bunch of foster puppies. Mommy is taking care
of them until they find their forever homes. Please consider adopting one of
them, so they will stop stealing my toys!

The Foster Puppies

Our Mommy works with a rescue group named Posh Pets. Right now, Mommy has
five foster puppies who are looking for good homes. They are:

Tiffany

Bru
```



The Adventures of Chloe and Diesel About Us Hi! I'm Chloe! Picture of Chloe in bag That probably isn't the best picture of me. Here's a better one: Picture of Chloe lying
Lab mix who loves to meet new people, go on car rides, and get lots of petting. I live with my Mommy and my brother Diesel: Picture of Diesel sitting He's a hound mix wh
hunt R-A-B-B-I-T-S. I have to spell it out so he doesn't hear and go crazy. We love to go to the dog park, and we have lots of friends there. I'll tell you about them later. Die
also live with a bunch of foster puppies. Mommy is taking care of them until they find their forever homes. Please consider adopting one of them, so they will stop stealing
The Foster Puppies Our Mommy works with a rescue group named Posh Pets. Right now, Mommy has five foster puppies who are looking for good homes. They are: Tiffa
Guinness Rolo Callie

What Happened?

- First of all, browsers ignore line breaks in a source document
- Second, the browser can display text from the source document, but we haven't told it about the ***structure*** of the document
    - use HTML to add structure to the HTML document itself, and then to its content

---

HTML Elements

- An element (contents and its markup) is identified by a tag in the source text
    - tags consist of the element name inside angle brackets: < and >
        - text inside the angle brackets is not displayed
    - closing tag has the element name after a backslash (</elementname>)
- Every element has markup (start and end tags), but some elements are empty (they have no content)
- Capitalization of element names doesn't matter in HTML, but it does in the stricter XHTML standard, so we'll use that style

---

Opening tag

Content
(text and/or other HTML elements)

Closing tag
(starts with a /)

<elementname> Content here </elementname>

<h1>Sunshine Bistro</h1>

- First line is a document type declaration, not an HTML element
- Entire document is contained within an `html` element (the root element), which may not be contained in any other element
- Head contains descriptive information about the page (title, style sheet, etc.)
- `meta` elements provide information about the document itself
- `title` element is mandatory
- `body` element contains what we want to show up in the browser window

```
<!DOCTYPE html>

<html>
<meta charset="utf-8">
<title>Title goes here</title>
</head>

<body>
Page content goes here.
</body>

</html>
```

Step 3: Identify Text Elements

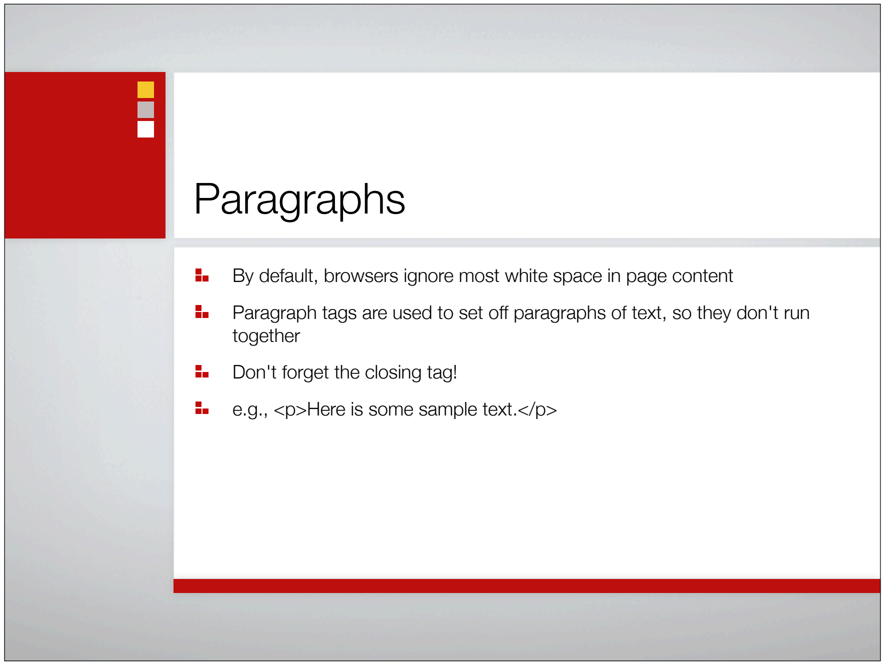
- Choose HTML elements that provide the most meaningful description of the content
  - this is called ***semantic markup***
- Markup gives the document structure by specifying relationships between elements
- This separation of meaning and structure vs. presentation instructions was not enforced so well in earlier versions of HTML
  - markup could be (and was) used to change a page's appearance: fonts, colors, alignment, etc.

Types of Text Elements

- Headings and subheadings
- Paragraphs
- Emphasized text
- Lists
- etc.

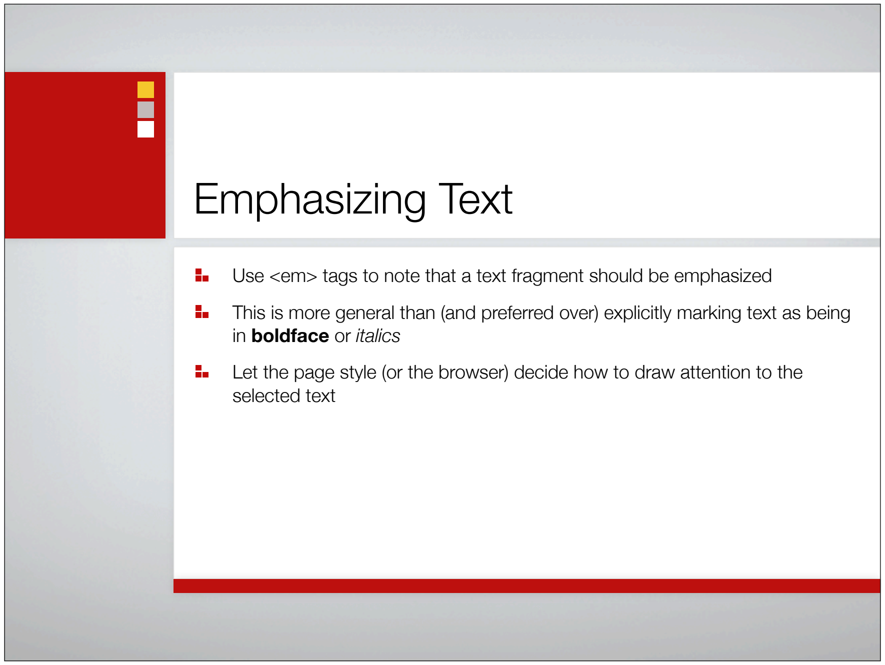
Headings

- Used to set off the sections of a page
- Six levels (h1 through h6)
  - h1 is the largest/most important, h6 is the smallest/least important
- Tags surround the heading text
  - e.g., <h1>Main Heading</h1>



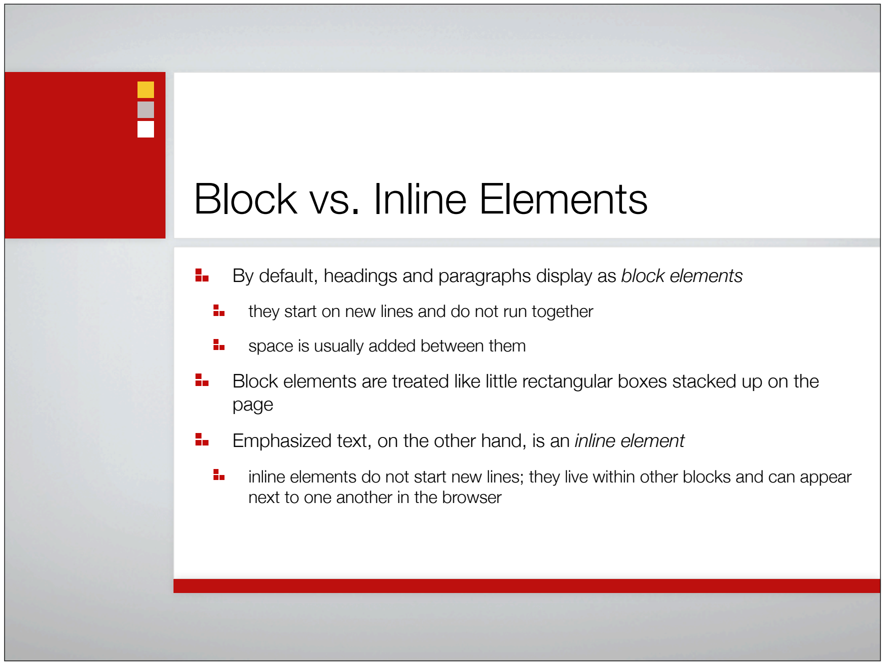
Paragraphs

- By default, browsers ignore most white space in page content
- Paragraph tags are used to set off paragraphs of text, so they don't run together
- Don't forget the closing tag!
- e.g., <p>Here is some sample text.</p>



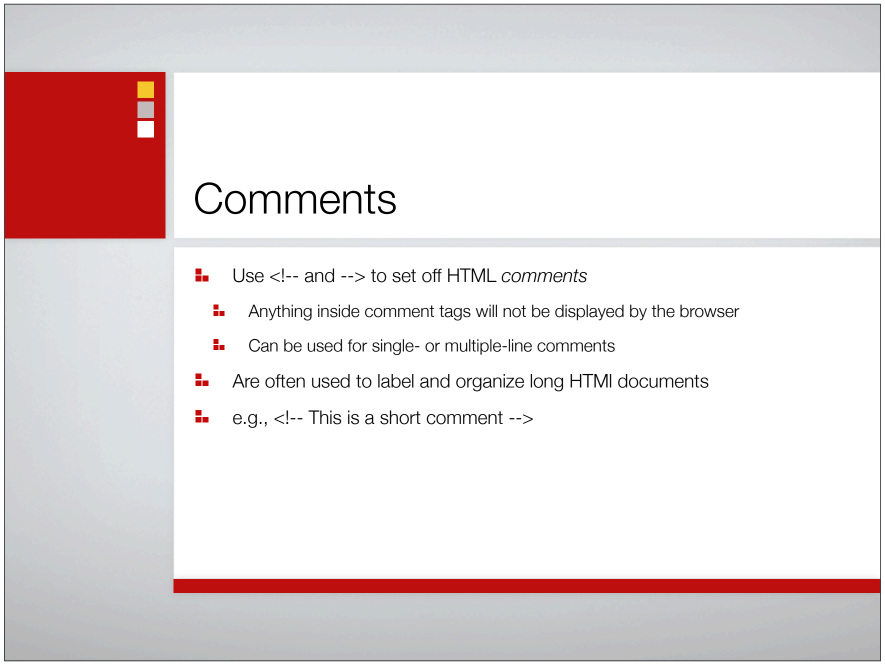
Emphasizing Text

- Use <em> tags to note that a text fragment should be emphasized
- This is more general than (and preferred over) explicitly marking text as being in **boldface** or *italics*
- Let the page style (or the browser) decide how to draw attention to the selected text



Block vs. Inline Elements

- By default, headings and paragraphs display as *block elements*
  - they start on new lines and do not run together
  - space is usually added between them
- Block elements are treated like little rectangular boxes stacked up on the page
- Emphasized text, on the other hand, is an *inline element*
  - inline elements do not start new lines; they live within other blocks and can appear next to one another in the browser

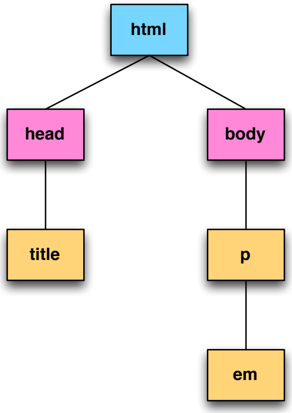


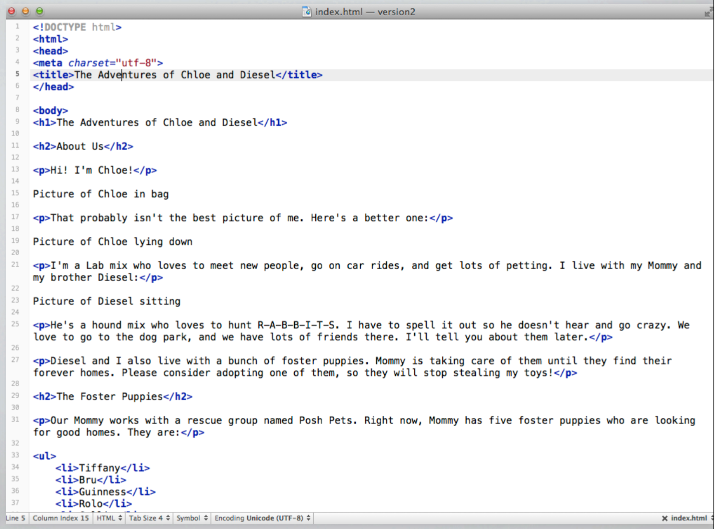
Comments

- Use <!-- and --> to set off HTML *comments*
  - Anything inside comment tags will not be displayed by the browser
  - Can be used for single- or multiple-line comments
- Are often used to label and organize long HTMl documents
- e.g., <!-- This is a short comment -->

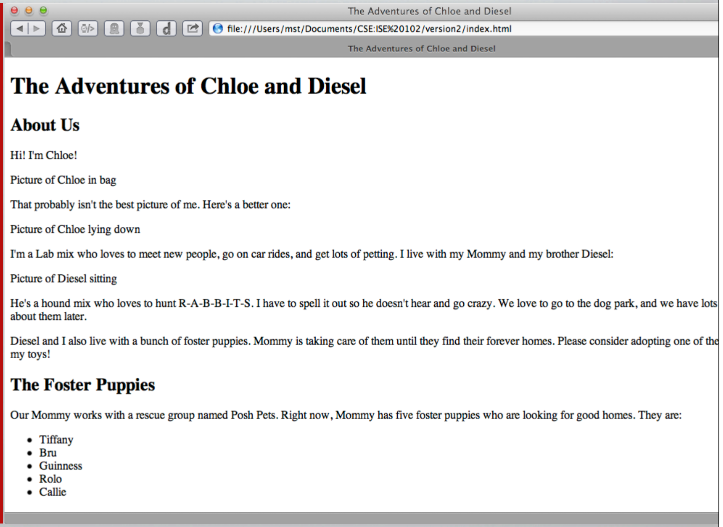
A Note on Nesting

- Tags may be nested inside one another to combine markup
  - e.g., using emphasized text inside a paragraph
- If you do this, you ***MUST*** close the inner/most recent tag before you close the outer tag
  - e.g., `<p>This is <em>some text.</em>.</p>` is OKAY
  - `<p>This is <em>some text.</p></em>` is NOT





```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>The Adventures of Chloe and Diesel</title>
</head>

<body>
<h1>The Adventures of Chloe and Diesel</h1>

<h2>About Us</h2>

<p>Hi! I'm Chloe!</p>

Picture of Chloe in bag

<p>That probably isn't the best picture of me. Here's a better one:</p>

Picture of Chloe lying down

<p>I'm a Lab mix who loves to meet new people, go on car rides, and get lots of petting. I live with my Mommy and
my brother Diesel:</p>

Picture of Diesel sitting

<p>He's a hound mix who loves to hunt R-A-B-B-I-T-S. I have to spell it out so he doesn't hear and go crazy. We
love to go to the dog park, and we have lots of friends there. I'll tell you about them later.</p>

<p>Diesel and I also live with a bunch of foster puppies. Mommy is taking care of them until they find their
forever homes. Please consider adopting one of them, so they will stop stealing my toys!</p>

<h2>The Foster Puppies</h2>

<p>Our Mommy works with a rescue group named Posh Pets. Right now, Mommy has five foster puppies who are looking
for good homes. They are:</p>

<ul>
    <li>Tiffany</li>
    <li>Bru</li>
    <li>Guinness</li>
    <li>Rolo</li>
```


The Adventures of Chloe and Diesel
file:///Users/mst/Documents/CSE:ISE%20102/version2/index.html
The Adventures of Chloe and Diesel

The Adventures of Chloe and Diesel

About Us

Hi! I'm Chloe!

Picture of Chloe in bag

That probably isn't the best picture of me. Here's a better one:

Picture of Chloe lying down

I'm a Lab mix who loves to meet new people, go on car rides, and get lots of petting. I live with my Mommy and my brother Diesel:

Picture of Diesel sitting

He's a hound mix who loves to hunt R-A-B-B-I-T-S. I have to spell it out so he doesn't hear and go crazy. We love to go to the dog park, and we have lots about them later.

Diesel and I also live with a bunch of foster puppies. Mommy is taking care of them until they find their forever homes. Please consider adopting one of the my toys!

The Foster Puppies

Our Mommy works with a rescue group named Posh Pets. Right now, Mommy has five foster puppies who are looking for good homes. They are:

- Tiffany
- Bru
- Guinness
- Rolo
- Callie

Default Styles

- The browser automatically attempts to give a Web page a visual hierarchy
  - uses larger/smaller text to indicate headings, etc.
- All browsers have built-in style sheets to do this
- If you don't like the look, create your own style sheet to override the browser default
  - DO NOT play with tag choices to attempt to format the page the way you want it to look
    - e.g., swapping <h1> tags for <h3> tags to get smaller headings

---

Step 4: Add an Image

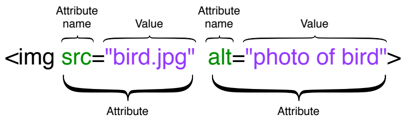
- We'll add images to the page in a moment
- First, we need to discuss *empty elements* and *attributes*
- An empty or *void* element has no text content; it just directs the browser to do something
- General syntax:

  `<element-name>`

- Examples: `<img>` (image elements) and `<br>` (line breaks)

Attributes

- Attributes are used to clarify or modify an element
  - they specify how it should be processed or displayed
  - e.g., the name of the image file to load (or its URL)



Attribute name

Value

Attribute name

Value

<img src="bird.jpg" alt="photo of bird">

Attribute

Attribute

- **Attribute names and values are separated by an equals sign (=)**
- **Multiple attributes are separated by a space**

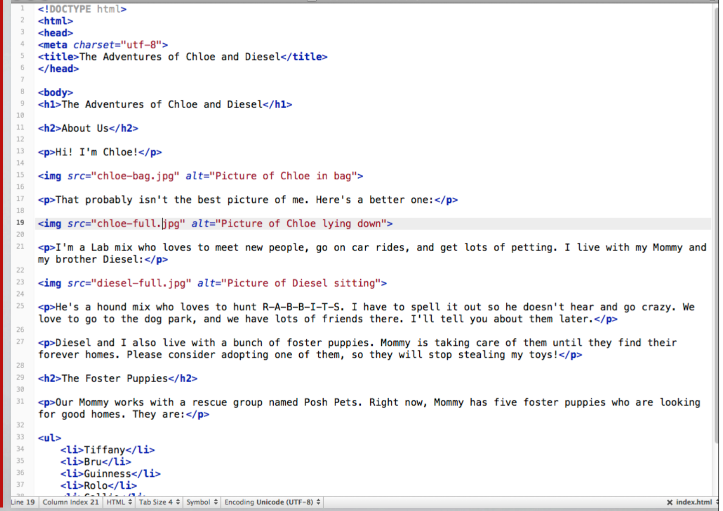
More on Attributes

- In non-empty elements, attributes ***ONLY*** go in the opening tag
- A value may be a number, a word, a string of text, a URL, or a measurement
- Some attributes are required (e.g., `src` in `<img>` tags)
- Attribute names are defined in the HTML specification
  - you can't make up or rename attributes

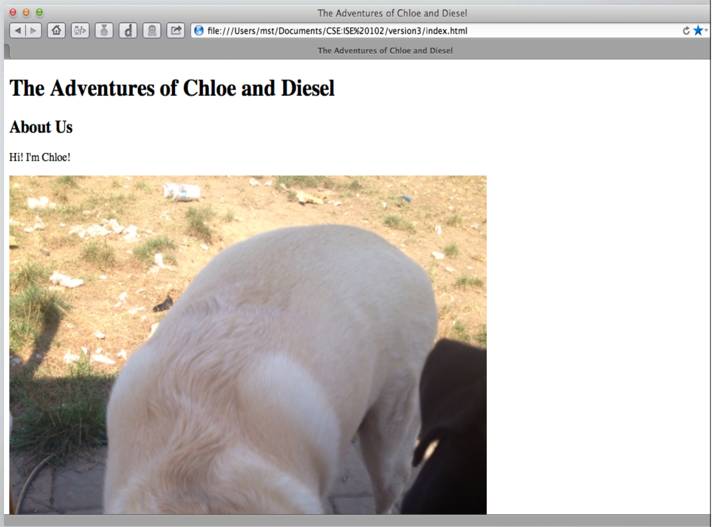
Using Image Tags

- The `<img>` tag has two required attributes:
  - `src` — the path to the image file to be loaded
  - `alt` — alternate text to show if the image cannot be displayed
- For now (and only for now) we'll put the images to display inside the same folder (and at the same level) as our HTML document
  - this means that we just put the image filename inside the `src` attribute

---



```html
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>The Adventures of Chloe and Diesel</title>
</head>

<body>
<h1>The Adventures of Chloe and Diesel</h1>

<h2>About Us</h2>

<p>Hi! I'm Chloe!</p>

<img src="chloe-bag.jpg" alt="Picture of Chloe in bag">

<p>That probably isn't the best picture of me. Here's a better one:</p>

<img src="chloe-full.jpg" alt="Picture of Chloe lying down">

<p>I'm a Lab mix who loves to meet new people, go on car rides, and get lots of petting. I live with my Mommy and my brother Diesel:</p>

<img src="diesel-full.jpg" alt="Picture of Diesel sitting">

<p>He's a hound mix who loves to hunt R-A-B-B-I-T-S. I have to spell it out so he doesn't hear and go crazy. We love to go to the dog park, and we have lots of friends there. I'll tell you about them later.</p>

<p>Diesel and I also live with a bunch of foster puppies. Mommy is taking care of them until they find their forever homes. Please consider adopting one of them, so they will stop stealing my toys!</p>

<h2>The Foster Puppies</h2>

<p>Our Mommy works with a rescue group named Posh Pets. Right now, Mommy has five foster puppies who are looking for good homes. They are:</p>

<ul>
    <li>Tiffany</li>
    <li>Bru</li>
    <li>Guinness</li>
    <li>Rolo</li>
```


The Adventures of Chloe and Diesel
file:///Users/mst/Documents/CSE:ISE%20102/version3/index.html
The Adventures of Chloe and Diesel

The Adventures of Chloe and Diesel

About Us

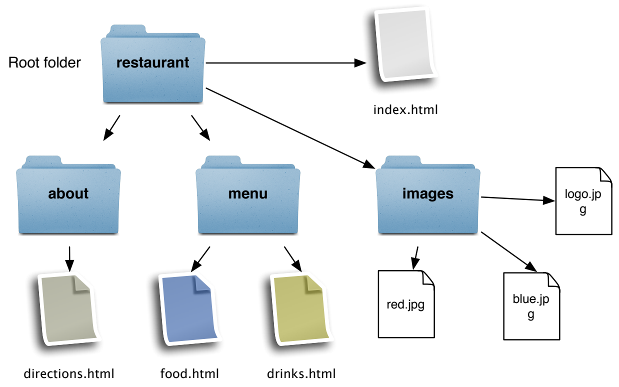
Hi! I'm Chloe!

Adjusting Image Size

- The browser will automatically determine the size of each image before displaying it
- Use the `width` and `height` attributes (with pixel values) to tell the browser how big each image is
  - this makes it easier to start laying out the page
- These attributes can also be used to (crudely) scale an image
  - don't do this if you can avoid it; resize the image in a separate application first
- `<img src="logo.png" width="200" height="250" alt="Our logo">`

Locating Images

- We can specify the location of an image file (or any resource) in one of two ways:
  - *relative addressing* tells us where the file is located relative to the current page
    - if we move the current document, the relative address can change!
  - *absolute addressing* gives us a non-changing address for the file/resource
    - works regardless of where we place or move the current document
- Either way, we use / and .. to move between folders/directories
  - `..` refers to the parent folder
  - / indicates a subfolder (e.g., myFolder/documents/index.html)



Path Addressing Examples

- Loading an image from index.html:
  - Refer to a file in the "images" subfolder
    - `<img src="images/logo.jpg" ...>`
- Loading an image from food.html:
  - Need to go up to parent folder first, and then down into the "images" folder
    - `<img src="../images/red.jpg" ...>`
- These are still relative paths
  - if we move files, the relative paths will all have to change

---

Absolute Paths

- Absolute paths begin with a forward slash (/) indicating the root of the Web site
- List the complete series of subfolders that you need to pass through in order to reach the desired file
  - e.g., `<img src="/images/logos/small-logo.jpg" ...>`
- As long as the resource itself doesn't move, you can move the current document relative to it without breaking the link

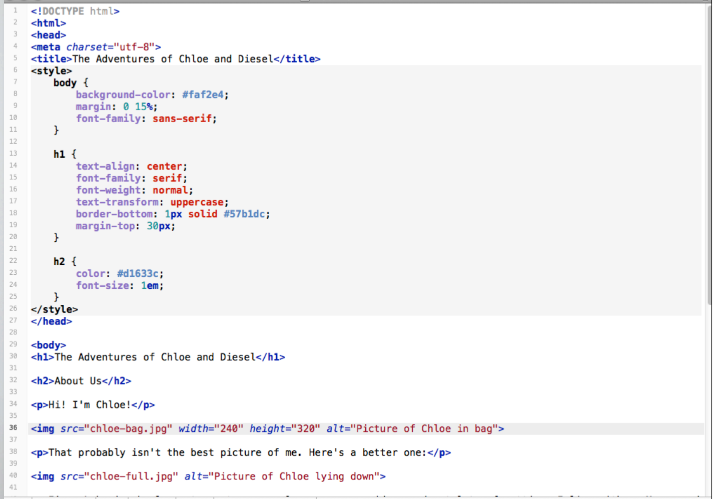
Image File Formats

Use JPEG for photos and complex graphics	Use PNG or GIF for images with solid colors, logos, and geometric shapes	
Works best for photographs	Works best for images with a few solid colors and images with lines	
Up to 16 million colors	PNG supports millions of different colors	GIF: limited to 256 colors
"Lossy" image compression	Lossless, but compresses files	Lossless
No transparency	Allows multiple transparent colors	Allows a single transparent color
No animation support	Animation through unsupported APNG format	Support for animations

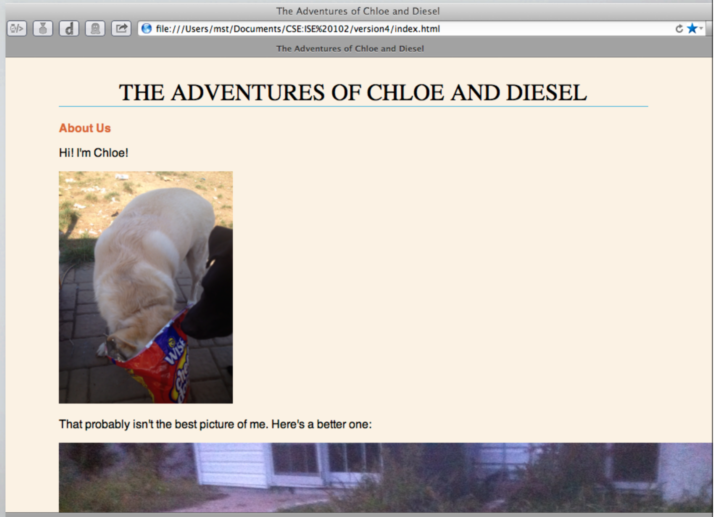
---

Step 5: Add a Style Sheet

- The `<style>` element can be used to apply a simple embedded style sheet to a Web page
  - this is one of several ways to add style sheets
- `<style>` goes in the document's head



```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>The Adventures of Chloe and Diesel</title>
<style>
    body {
        background-color: #faf2e4;
        margin: 0 15%;
        font-family: sans-serif;
    }

    h1 {
        text-align: center;
        font-family: serif;
        font-weight: normal;
        text-transform: uppercase;
        border-bottom: 1px solid #57b1dc;
        margin-top: 30px;
    }

    h2 {
        color: #d1633c;
        font-size: 1em;
    }
</style>
</head>

<body>
<h1>The Adventures of Chloe and Diesel</h1>

<h2>About Us</h2>

<p>Hi! I'm Chloe!</p>

<img src="chloe-bag.jpg" width="240" height="320" alt="Picture of Chloe in bag">

<p>That probably isn't the best picture of me. Here's a better one:</p>

<img src="chloe-full.jpg" alt="Picture of Chloe lying down">
```


The Adventures of Chloe and Diesel
file:///Users/mst/Documents/CSE:ISE%20102/version4/index.html
The Adventures of Chloe and Diesel

THE ADVENTURES OF CHLOE AND DIESEL

About Us

Hi! I'm Chloe!

That probably isn't the best picture of me. Here's a better one:

New Body Style

```
body {
        background-color: #faf2e4;
        margin: 0 15%;
        font-family: sans-serif;
}
```

New Body Style

```
body {
        background-color: #faf2e4;
        margin: 0 15%;
        font-family: sans-serif;
}
```

New Header (1) Style

```
h1 {
    text-align: center;
    font-family: serif;
    font-weight: normal;
    text-transform: uppercase;
    border-bottom: 1px solid #57b1dc;
    margin-top: 30px;
}
```

New Header (2) Style

```
h2 {
    color: #d1633c;
    font-size: 1em;
}
```

What Just Happened?!

- We defined a style sheet that told the browser how to display certain types of text elements
  - these styles override the ones that are built-in to the browser
- For example, the body section set the page's background color, margins, and typeface
  - we'll go into all of the details of CSS later this semester

When Things Go Wrong...

- Writing HTML markup by hand can be error-prone
  - missing slashes or quote marks, or even whole tags
- Omitting a slash in a closing tag can cause the current formatting to continue on to the end of the document
- Omitting a bracket can cause the browser to not display part or all of your page
  - the browser assumes that everything up to the next closing bracket is part of the original HTML tag

Validation

- Most browsers will work more-or-less correctly, even if a page has errors
- Use a ***validator*** to check your code for errors and to make sure that you're abiding by the rules of the HTML version you're using
  - e.g., http://validator.w3c.org
  - validators look for mismatched tags, nesting errors, required rules and attributes, etc.
- Valid documents display more consistently on different browsers, they may display more quickly, and they are more accessible

Next Time

- More HTML markup
  - Lists
  - Hyperlinks
- Notes on XHTML