COSC 360
Web Development

Topic 2: Intro to HTML

Chapter 3

1 What Is HTML and Where Did It Come from?

2 HTML Syntax

3 Semantic Markup

4 Structure of HTML Documents

5 Quick Tour of HTML Elements

6 HTML5 Semantic Structure Elements

7 Summary

Brief History of HTML

Did we mention that this will be brief?

- HTML stands for *Hyper Text Markup Language*
- Initially specified in 1991 by Tim Berners-Lee, and codified by the World-Wide Web Consortium (better known as the **W3C**) in 1997.
  - HTML 2 : 1995
  - HTML 3 : 1997
  - HTML 4 : 1997
  - **HTML 5 : 2014**

HTML Syntax

What is a markup language?

HTML is defined as a **markup language**.

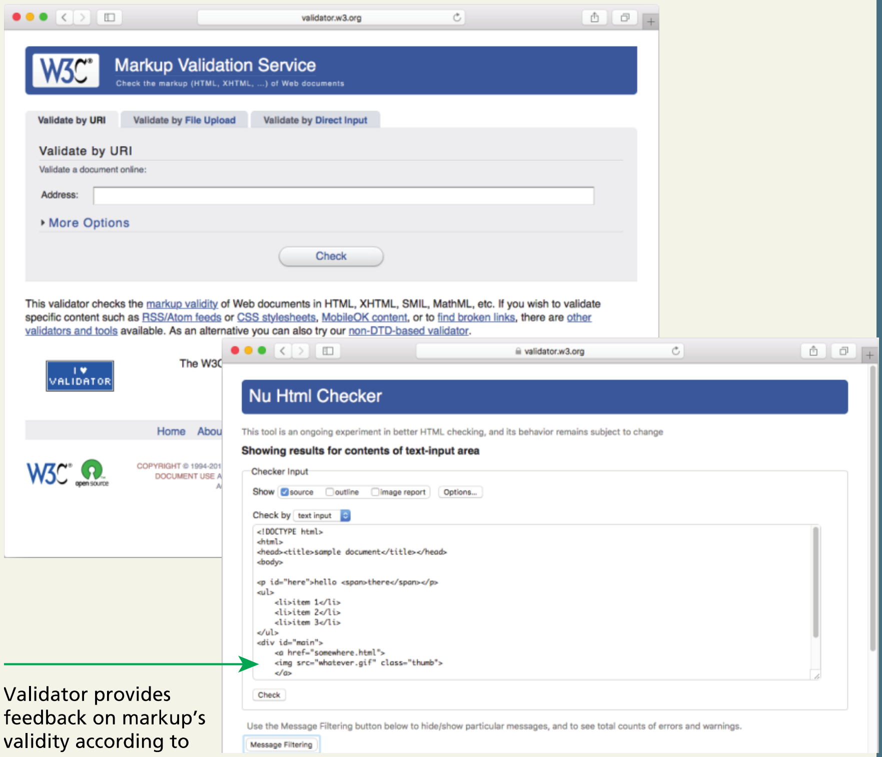
- Markup is a way of annotating a document in such a way to make the annotations distinct from the text being annotated.
- HTML has been through many versions and branches, the details of which might matter if you ever see old HTML code.
- The term comes from the days of print, when editors would write instructions on manuscript pages that might be revision instructions to the author or copy editor.

Markup

What is it again?

At its simplest, **markup** is a way to indicate *information about the content*

- This “information about content” in HTML is implemented via **tags** (aka elements).
- The markup in the previous slide consists of the red text and the various circles and arrows on the one page, and the little yellow sticky notes on the other.
- HTML does the same thing but uses textual tags.



This validator checks the markup validity of Web documents in HTML, XHTML, SMIL, MathML, etc. If you wish to validate specific content such as RSS/Atom feeds or CSS stylesheets, MobileOK content, or to find broken links, there are other validators and tools available. As an alternative you can also try our non-DTD-based validator.

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

```
<!DOCTYPE html>
<html>
<head><title>sample document</title></head>
<body>

<p id="here">hello <span>there</span></p>
<ul>
    <li>item 1</li>
    <li>item 2</li>
    <li>item 3</li>
</ul>
<div id="main">
    <a href="somewhere.html">
    <img src="whatever.gif" class="thumb">
    </a>
```

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Validator provides feedback on markup’s validity according to

HTML5

Three main aims

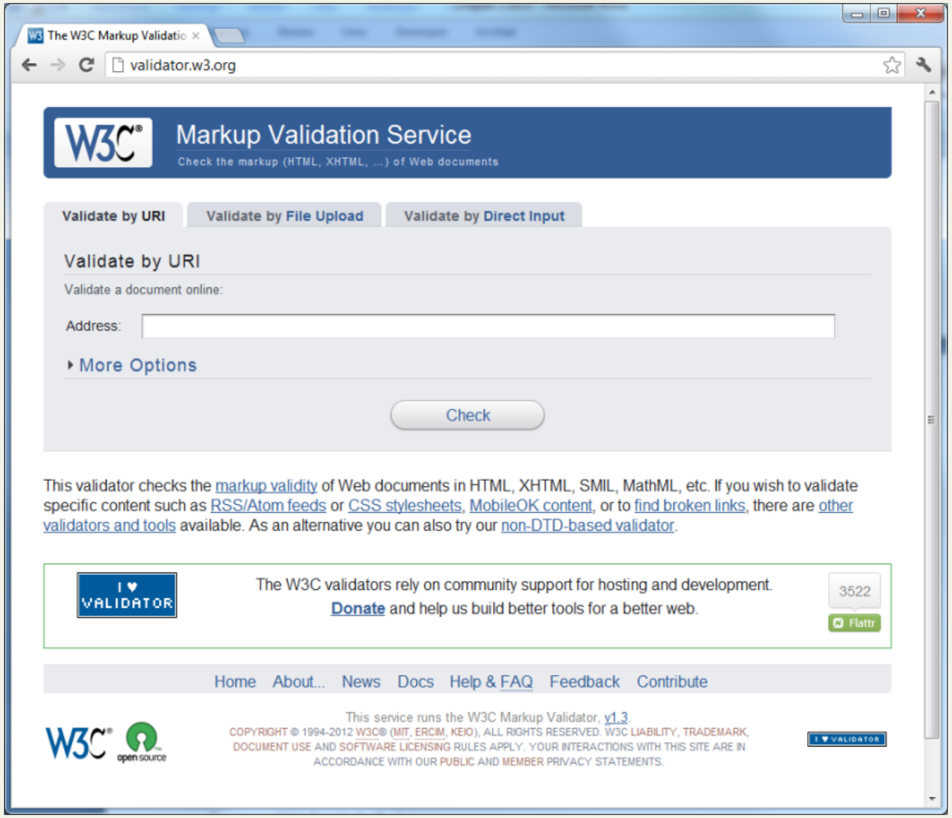
There are three main aims to HTML5:

- Specify unambiguously how browsers should deal with invalid markup.
- Provide an open, non-proprietary programming framework (via Javascript) for creating rich web applications.
- Be backwards compatible with the existing web.

Validators

How to ensure your pages follow a standard

A key part of the standards movement in the web development community of the 2000s was the use of **HTML Validators** as a means of verifying that a web page's markup followed the rules for XHTML transitional or strict.



Validate by URI

Validate a document online:

This validator checks the markup validity of Web documents in HTML, XHTML, SMIL, MathML, etc. If you wish to validate specific content such as RSS/Atom feeds or CSS stylesheets, MobileOK content, or to find broken links, there are other validators and tools available. As an alternative you can also try our non-DTD-based validator.

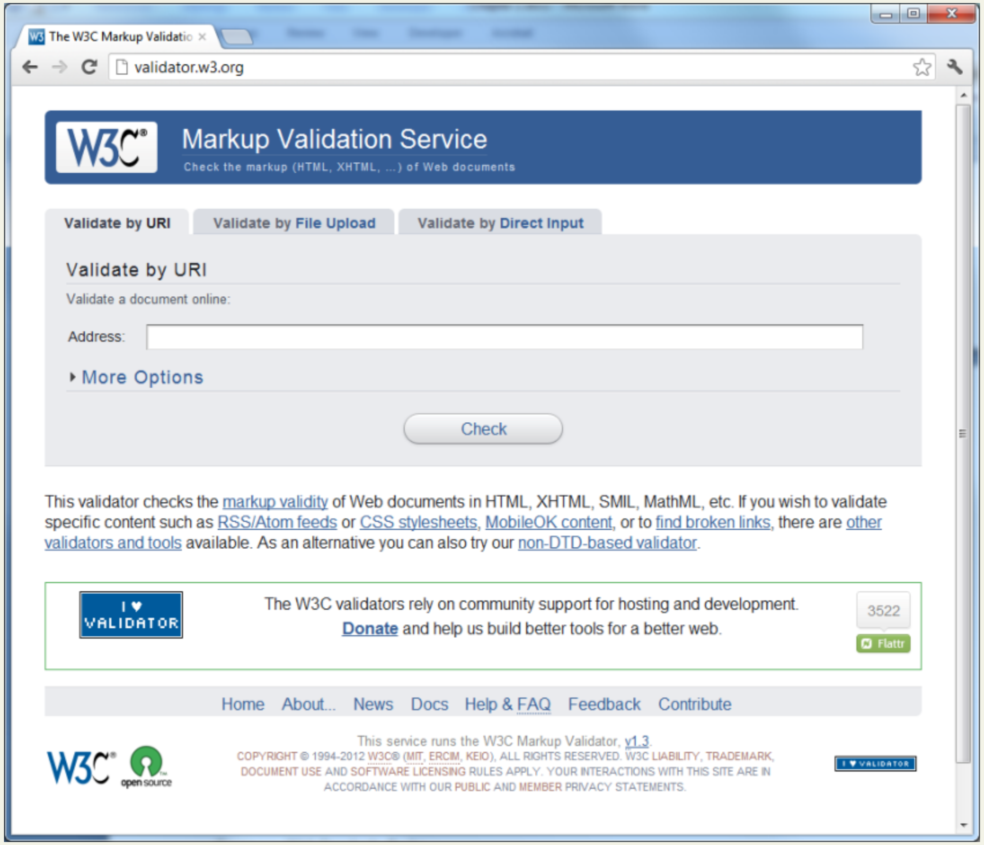
The W3C validators rely on community support for hosting and development. Donate and help us build better tools for a better web.

This service runs the W3C Markup Validator, v1.3.

COPYRIGHT © 1994-2012 W3C® (MIT, ERCIM, KEIO), ALL RIGHTS RESERVED. W3C LIABILITY, TRADEMARK, DOCUMENT USE AND SOFTWARE LICENSING RULES APPLY. YOUR INTERACTIONS WITH THIS SITE ARE IN ACCORDANCE WITH OUR PUBLIC AND MEMBER PRIVACY STATEMENTS.

How about an example

Only if you have an internet connection



Markup Validation Service

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

Validate by URI

Validate a document online:

Address:

This validator checks the markup validity of Web documents in HTML, XHTML, SMIL, MathML, etc. If you wish to validate specific content such as RSS/Atom feeds or CSS stylesheets, MobileOK content, or to find broken links, there are other validators and tools available. As an alternative you can also try our non-DTD-based validator.

The W3C validators rely on community support for hosting and development. Donate and help us build better tools for a better web.

This service runs the W3C Markup Validator, v1.3.

COPYRIGHT © 1994-2012 W3C® (MIT, ERCIM, KEIO), ALL RIGHTS RESERVED. W3C LIABILITY, TRADEMARK, DOCUMENT USE AND SOFTWARE LICENSING RULES APPLY. YOUR INTERACTIONS WITH THIS SITE ARE IN ACCORDANCE WITH OUR PUBLIC AND MEMBER PRIVACY STATEMENTS.

Try on your own:
Open a web browser to the W3C validator and find a few websites to test.

https://validator.w3.org

You can try using http://www.ubc.ca

Chapter 3

1 What Is HTML and Where Did It Come from?

2 HTML Syntax

3 Semantic Markup

4 Structure of HTML Documents

5 Quick Tour of HTML Elements

6 HTML5 Semantic Structure Elements

7 Summary

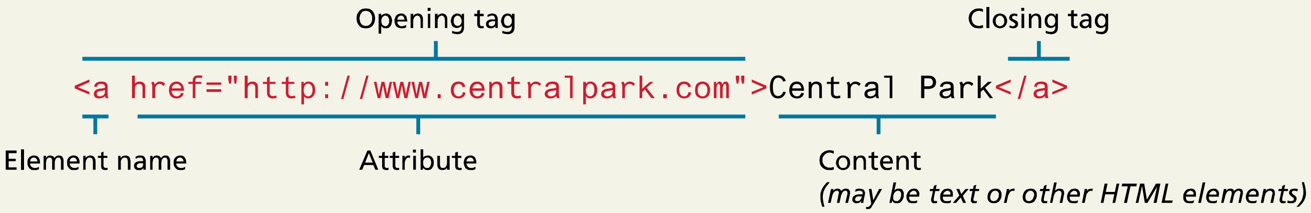
Elements and Attributes

More syntax

HTML documents are composed of textual content and **HTML element** .

HTML element encompasses

- the **element name** within angle brackets (i.e., the **tag**) and
- HTML elements can also contain **attributes**.
- **the content** within the tag.



Opening tag

Closing tag

```
<a href="http://www.centralpark.com">Central Park</a>
```

Element name

Attribute

Content

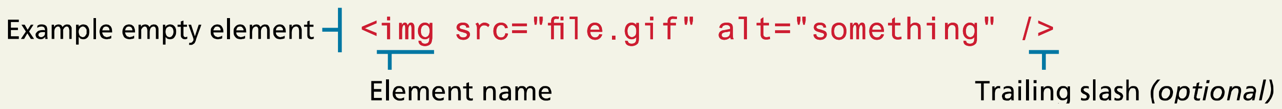
(may be text or other HTML elements)

HTML Syntax

Elements and Attributes

An **empty element** does not contain any text content; instead, it is an instruction to the browser to do something.

- In XHTML, empty elements had to be terminated by a trailing slash.
- In HTML5, the trailing slash in empty elements is optional.



```
<img src="file.gif" alt="something" />
```

Nesting HTML elements

Often an HTML element will contain other HTML elements.

In such a case, the container element is said to be a parent of the contained, or child, element.

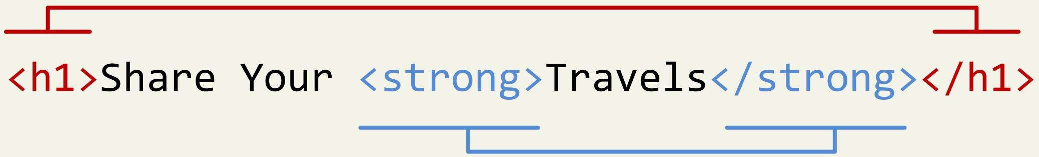
Any elements contained within the child are said to be **descendents** of the parent element; likewise, any given child element, may have a variety of **ancestors.**

Nesting HTML elements

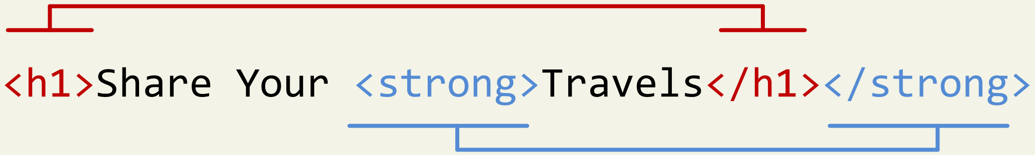
In order to properly construct a hierarchy of elements, your browser expects each HTML nested element to be properly nested.

That is, a child’s ending tag must occur before its parent’s ending tag.

Correct Nesting



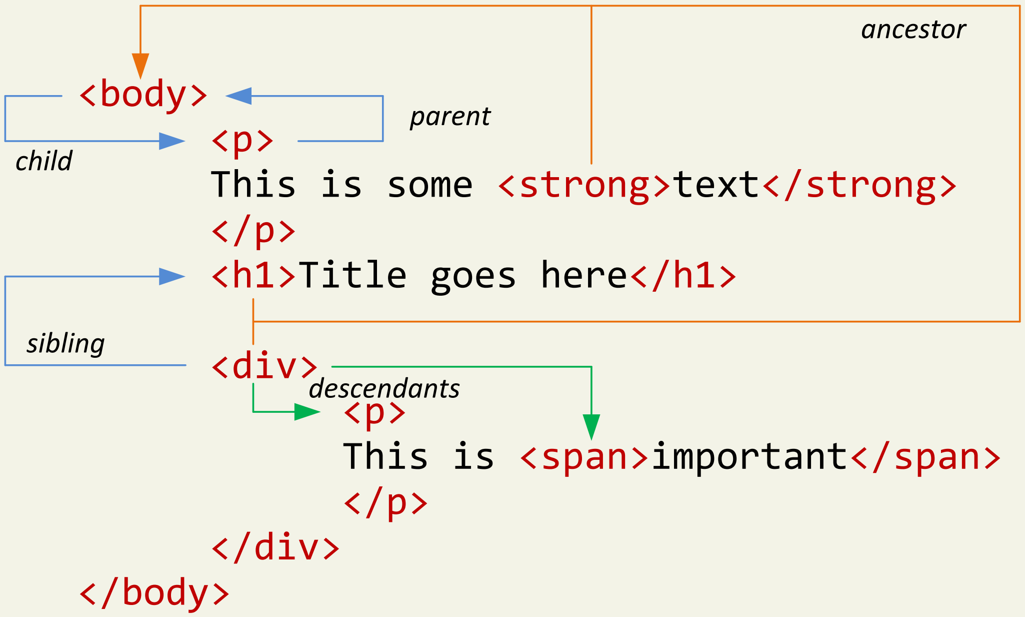
```
<h1>Share Your <strong>Travels</strong></h1>
```



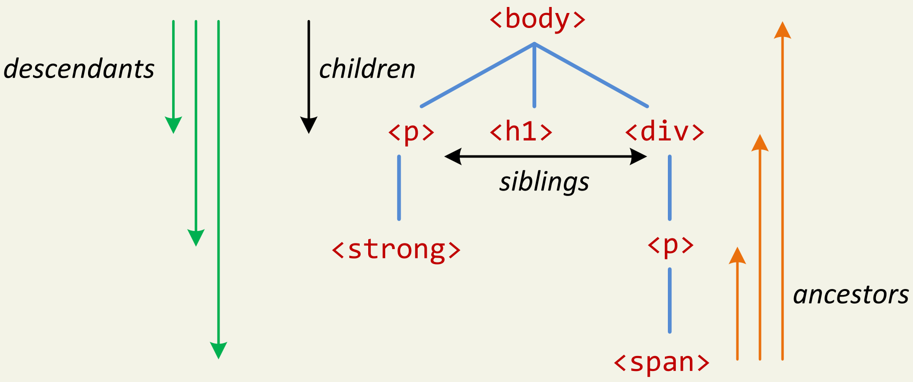
```
<h1>Share Your <strong>Travels</h1></strong>
```

Incorrect Nesting

Hierarchy of elements



```
<body>
   <p>
   This is some <strong>text</strong>
   </p>
   <h1>Title goes here</h1>
   <div>
      <p>
      This is <span>important</span>
      </p>
   </div>
</body>
```



Question

Consider the following element:

<a href=“https://www.funwebdev.com”></a>

In this case, href is a…

A. Element

B. Attribute

C. Content(s)

D. Tag

Question

Consider the following element:

<a href=“https://www.funwebdev.com”></a>

In this case, href is a…

A. Element

B. Attribute

C. Content(s)

D. Tag

Chapter 3

1 What Is HTML and Where Did It Come from?

2 HTML Syntax

3 Semantic Markup

4 Structure of HTML Documents

5 Quick Tour of HTML Elements

6 HTML5 Semantic Structure Elements

7 Summary

Semantic Markup

What does it mean?

Over the past decade, a strong and broad consensus has grown around the belief that HTML documents should **only** focus on the structure of the document.

Information about how the content should look when it is displayed in the browser is best left to CSS (Cascading Style Sheets).

Semantic Markup

As a consequence, beginning HTML authors are often counseled to create **semantic HTML** documents.

That is, an HTML document should not describe how to visually present content, but only describe its content’s structural semantics or meaning.

Structure

Structure is a vital way of communicating information in paper and electronic documents.

All of the tags that we will examine in this presentation are used to describe the basic structural information in a document, such as articles, headings, lists, paragraphs, links, images, navigation, footers, and so on.

Semantic Markup

Its advantages

Eliminating presentation-oriented markup and writing semantic HTML markup has a variety of important advantages:

Maintainability. Semantic markup is easier to update and change than web pages that contain a great deal of presentation markup.

Faster. Semantic web pages are typically quicker to author and faster to download.

Accessibility. Visiting a web page using voice reading software can be a very frustrating experience if the site does not use semantic markup.

Search engine optimization. Semantic markup provides better instructions for search engines: it tells them what things are important content on the site.

Chapter 3

1 What Is HTML and Where Did It Come from?

2 HTML Syntax

3 Semantic Markup

4 Structure of HTML Documents

5 Quick Tour of HTML Elements

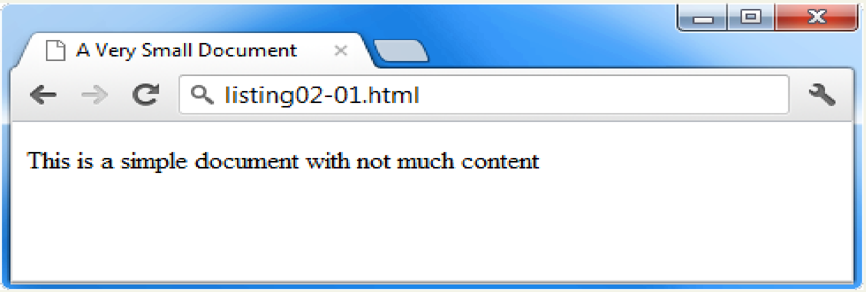
6 HTML5 Semantic Structure Elements

7 Summary

Simplest HTML document

1

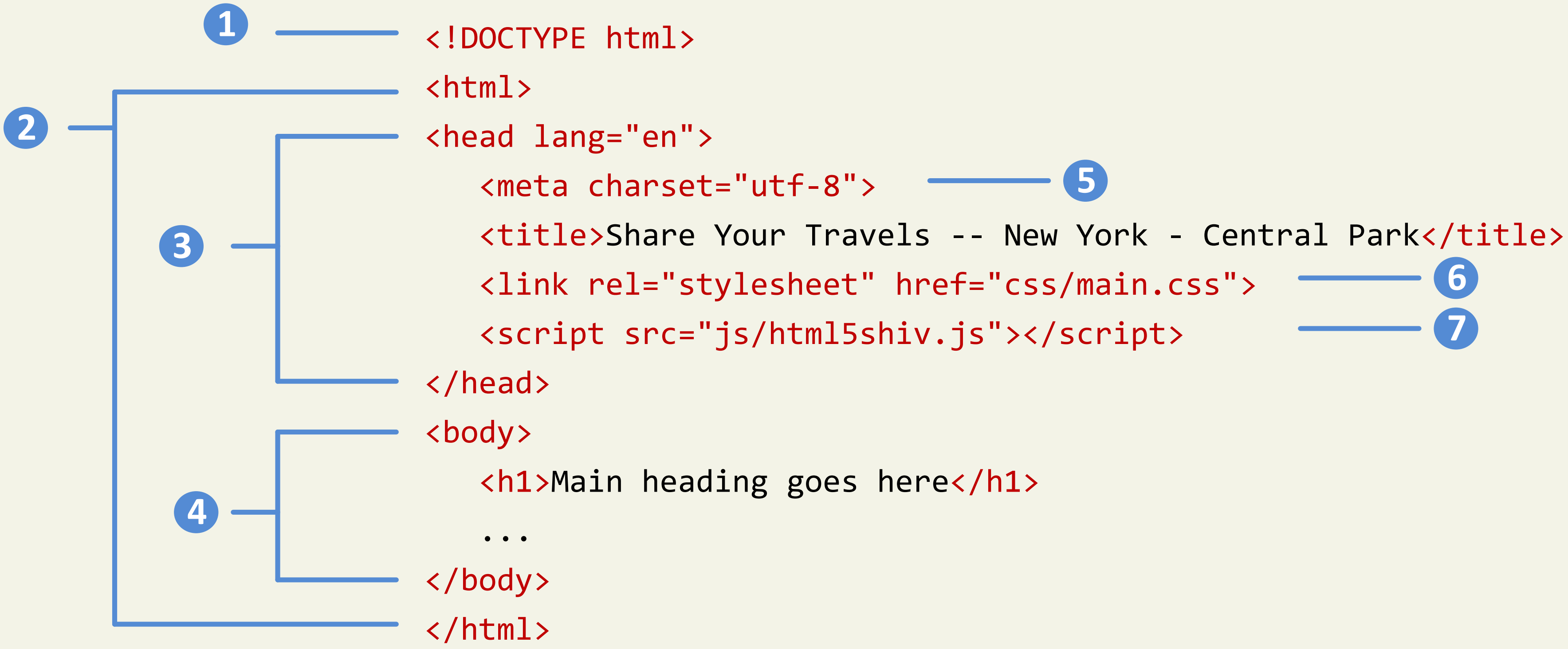
```
<!DOCTYPE html>
<title>A Very Small Document</title>
<p>This is a simple document with not much content</p>
```



This is a simple document with not much content

The <title> element (Item 1) is used to provide a broad description of the content. The title is not displayed within the browser window. Instead, the title is typically displayed by the browser in its window and/or tab.

A More Complete Document



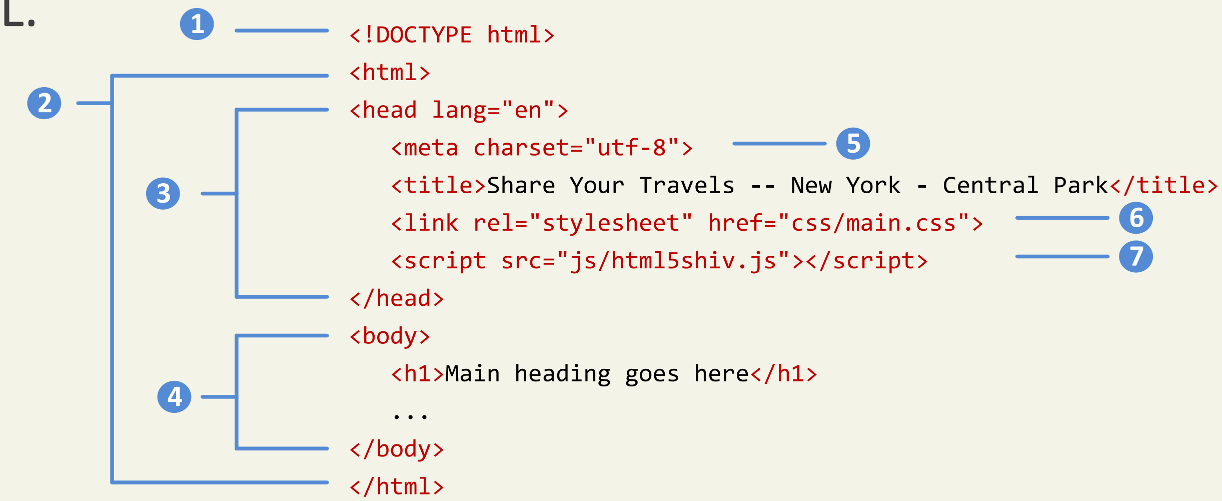
```
<!DOCTYPE html>
<html>
<head lang="en">
   <meta charset="utf-8">
   <title>Share Your Travels -- New York - Central Park</title>
   <link rel="stylesheet" href="css/main.css">
   <script src="js/html5shiv.js"></script>
</head>
<body>
   <h1>Main heading goes here</h1>
   ...
</body>
</html>
```

1 DOCTYPE

(short for **Document Type Definition**)

Tells the browser (or any other client software that is reading this HTML document) what type of document it is about to process.

Notice that it does not indicate what version of HTML is contained within the document: it only specifies that it contains HTML.



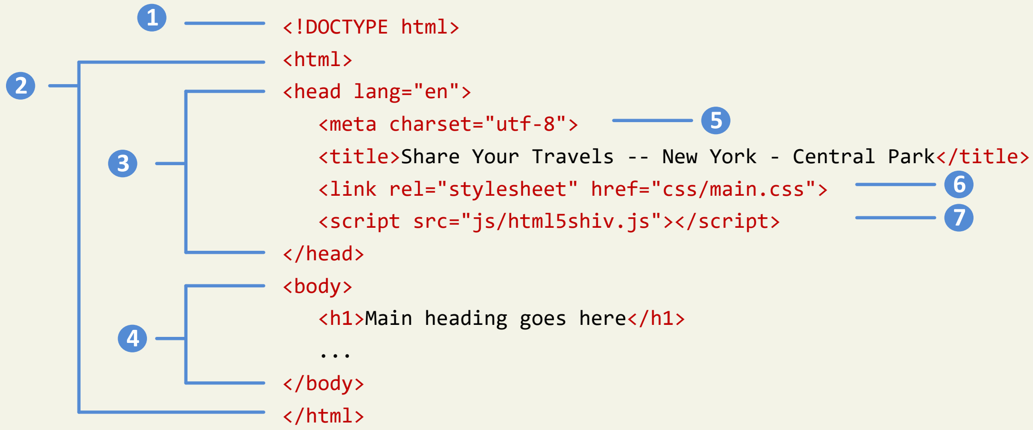
```
<!DOCTYPE html>
<html>
<head lang="en">
   <meta charset="utf-8">
   <title>Share Your Travels -- New York - Central Park</title>
   <link rel="stylesheet" href="css/main.css">
   <script src="js/html5shiv.js"></script>
</head>
<body>
   <h1>Main heading goes here</h1>
   ...
</body>
</html>
```

HTML, Head, and Body

HTML5 does not require the use of the <html>, <head>, and <body>.

However, in XHTML they were required, and most web authors continue to use them.

2 The <html> element is sometimes called the **root element** as it contains all the other HTML elements in the document.



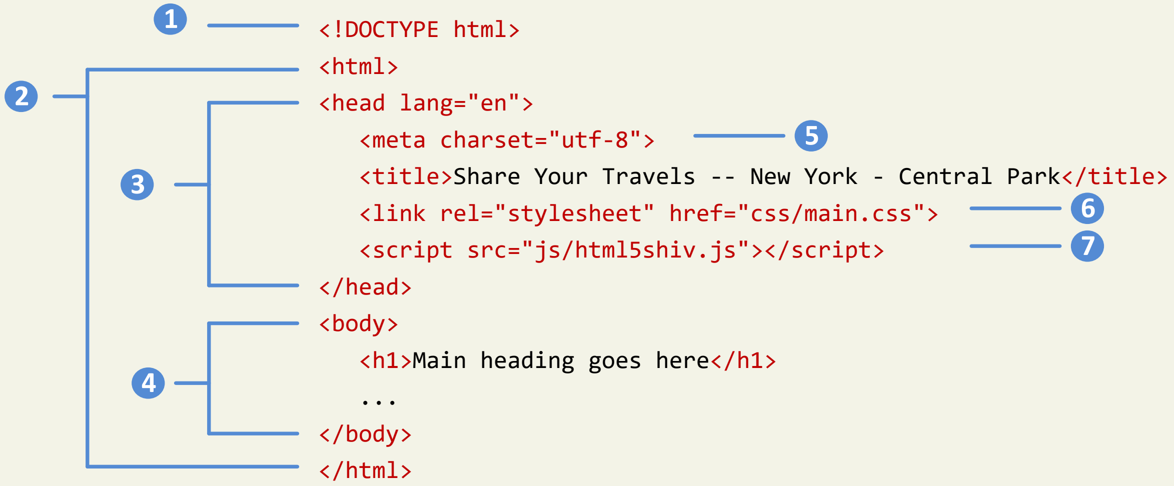
```
<!DOCTYPE html>
<html>
<head lang="en">
   <meta charset="utf-8">
   <title>Share Your Travels -- New York - Central Park</title>
   <link rel="stylesheet" href="css/main.css">
   <script src="js/html5shiv.js"></script>
</head>
<body>
   <h1>Main heading goes here</h1>
   ...
</body>
</html>
```

Head and Body

HTML pages are divided into two sections: the **head** and the **body**, which correspond to the <head> and <body> elements.

3 The head contains descriptive elements *about* the document

4 The body contains content that will be displayed by the browser.



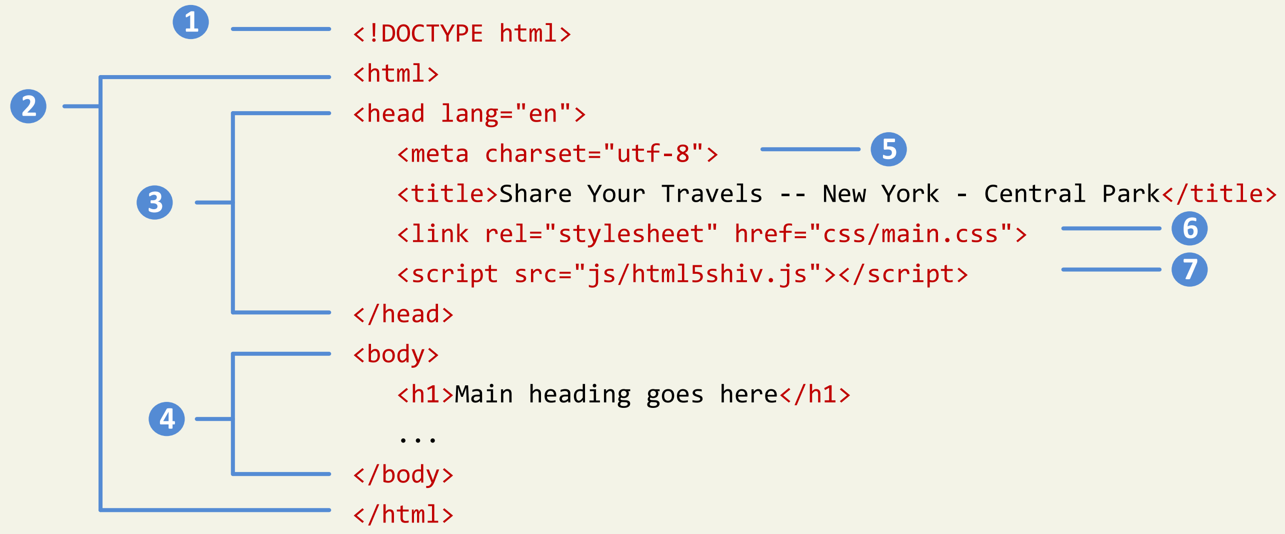
```
<!DOCTYPE html>
<html>
<head lang="en">
   <meta charset="utf-8">
   <title>Share Your Travels -- New York - Central Park</title>
   <link rel="stylesheet" href="css/main.css">
   <script src="js/html5shiv.js"></script>
</head>
<body>
   <h1>Main heading goes here</h1>
   ...
</body>
</html>
```

Inside the head

There are no brains

You will notice that the <head> element contains a variety of additional elements.

5 The first of these is the <meta> element. Our example declares that the character encoding for the document is UTF-8.



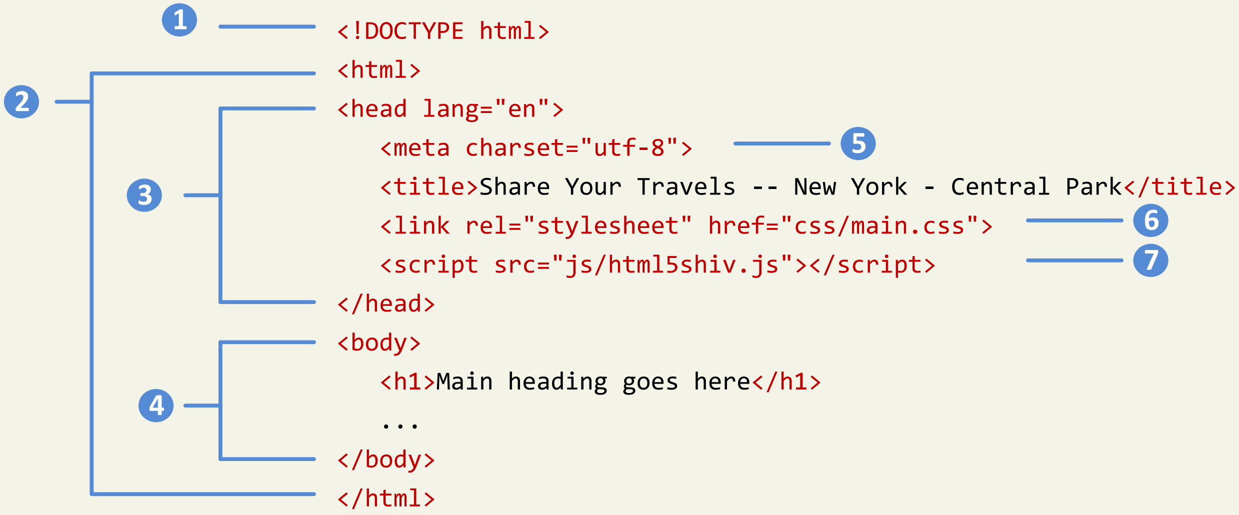
```
<!DOCTYPE html>
<html>
<head lang="en">
   <meta charset="utf-8">
   <title>Share Your Travels -- New York - Central Park</title>
   <link rel="stylesheet" href="css/main.css">
   <script src="js/html5shiv.js"></script>
</head>
<body>
   <h1>Main heading goes here</h1>
   ...
</body>
</html>
```

Inside the head

No brains but metas, styles and javascripts

6 Our example specifies an external CSS style sheet file that is used with this document.

7 It also references an external Javascript file.



```
<!DOCTYPE html>
<html>
<head lang="en">
   <meta charset="utf-8">
   <title>Share Your Travels -- New York - Central Park</title>
   <link rel="stylesheet" href="css/main.css">
   <script src="js/html5shiv.js"></script>
</head>
<body>
   <h1>Main heading goes here</h1>
   ...
</body>
</html>
```

Chapter 3

1 What Is HTML and Where Did It Come from?

2 HTML Syntax

3 Semantic Markup

4 Structure of HTML Documents

5 Quick Tour of HTML Elements

6 HTML5 Semantic Structure Elements

7 Summary

Why a Quick Tour?

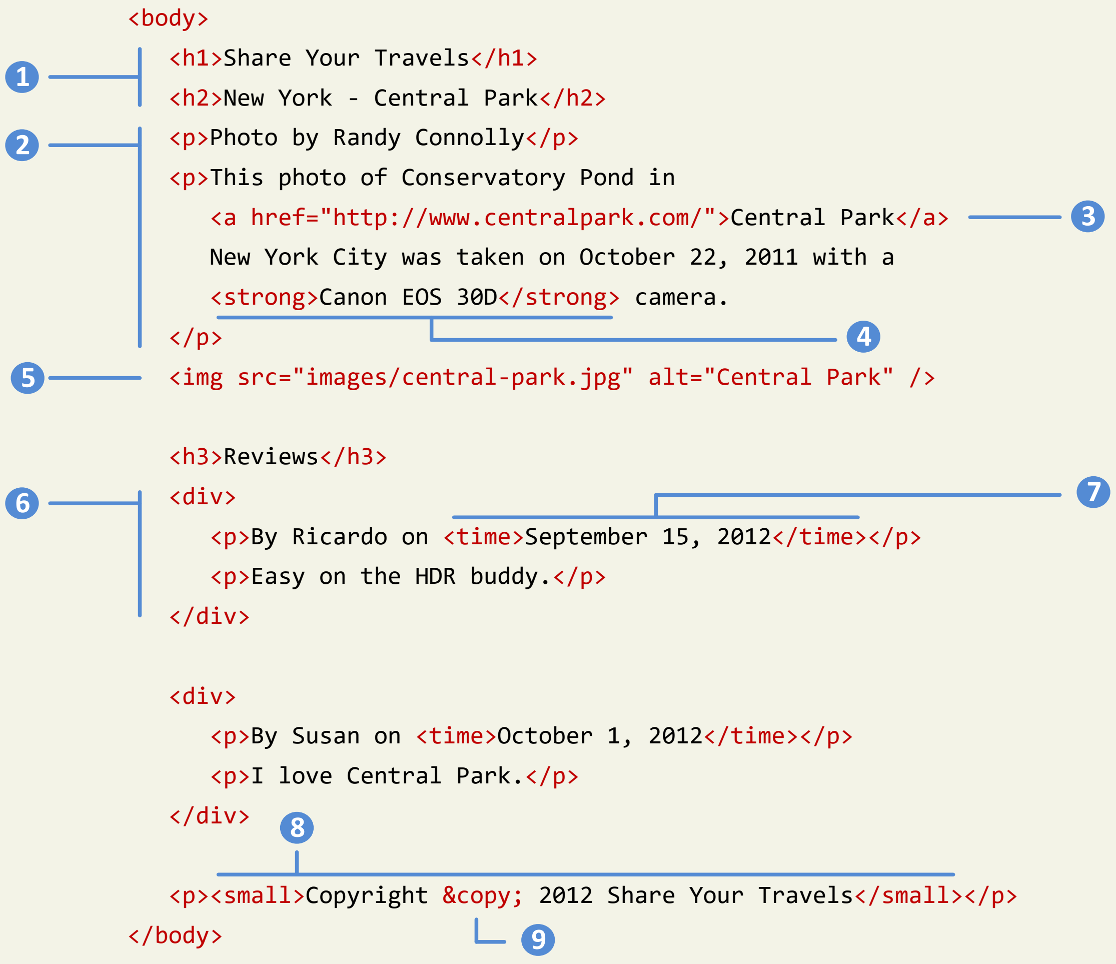
HTML5 contains many structural and presentation elements – too many to completely cover in this presentation.

Rather than comprehensively cover all these elements, this presentation will provide a quick overview of the most common elements.

The Resource:

http://www.w3schools.com/html/default.asp

Sample Document

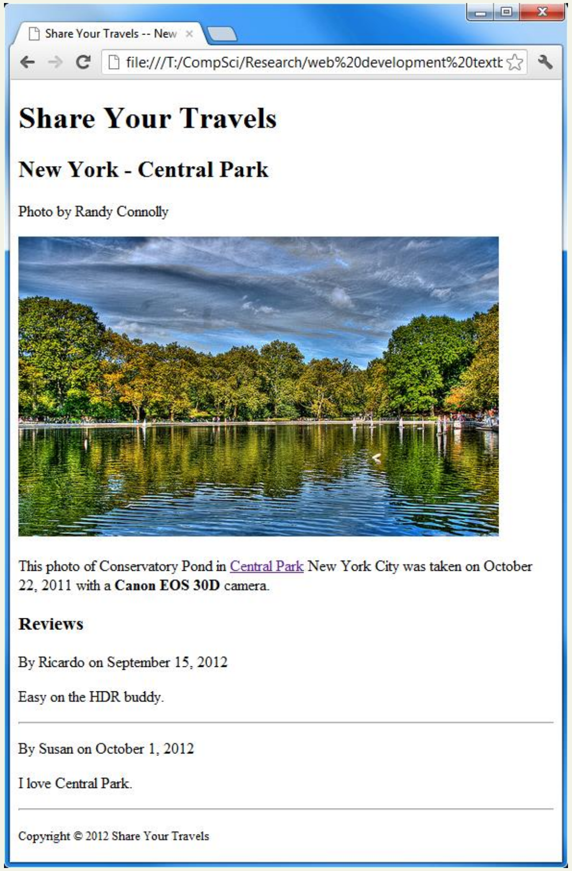


```
<body>
   <h1>Share Your Travels</h1>
   <h2>New York - Central Park</h2>
   <p>Photo by Randy Connolly</p>
   <p>This photo of Conservatory Pond in
      <a href="http://www.centralpark.com/">Central Park</a>
      New York City was taken on October 22, 2011 with a
      <strong>Canon EOS 30D</strong> camera.
   </p>
   <img src="images/central-park.jpg" alt="Central Park" />

   <h3>Reviews</h3>
   <div>
      <p>By Ricardo on <time>September 15, 2012</time></p>
      <p>Easy on the HDR buddy.</p>
   </div>

   <div>
      <p>By Susan on <time>October 1, 2012</time></p>
      <p>I love Central Park.</p>
   </div>

   <p><small>Copyright &copy; 2012 Share Your Travels</small></p>
</body>
```



Share Your Travels -- New
file:///T:/CompSci/Research/web%20development%20textb

Share Your Travels

New York - Central Park

Photo by Randy Connolly

This photo of Conservatory Pond in Central Park New York City was taken on October 22, 2011 with a **Canon EOS 30D** camera.

Reviews

By Ricardo on September 15, 2012

Easy on the HDR buddy.

By Susan on October 1, 2012

I love Central Park.

Copyright © 2012 Share Your Travels

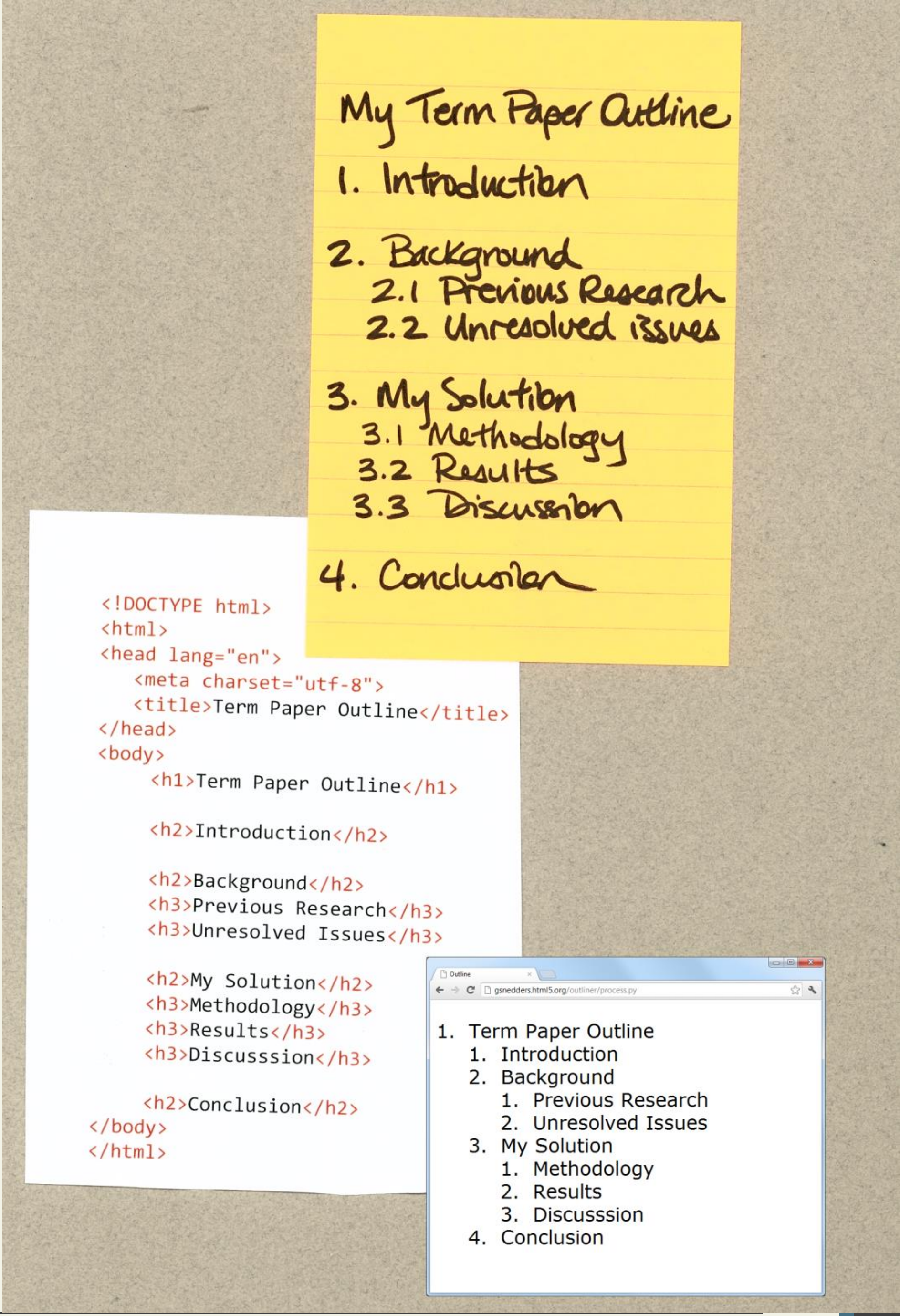
1 Headings

<h1>, <h2>, <h3>, etc

HTML provides six levels of heading (**h1, h2, h3**, …), with the higher heading number indicating a heading of less importance.

Headings are an essential way for document authors use to show their readers the structure of the document.

Headings are also used by the browser to create a document outline for the page.



My Term Paper Outline

1. Introduction
2. Background
   - 2.1 Previous Research
   - 2.2 Unresolved issues
3. My Solution
   - 3.1 Methodology
   - 3.2 Results
   - 3.3 Discussion
4. Conclusion

```
<!DOCTYPE html>
<html>
<head lang="en">
   <meta charset="utf-8">
   <title>Term Paper Outline</title>
</head>
<body>
     <h1>Term Paper Outline</h1>

     <h2>Introduction</h2>

     <h2>Background</h2>
     <h3>Previous Research</h3>
     <h3>Unresolved Issues</h3>

     <h2>My Solution</h2>
     <h3>Methodology</h3>
     <h3>Results</h3>
     <h3>Discusssion</h3>

     <h2>Conclusion</h2>
</body>
</html>
```

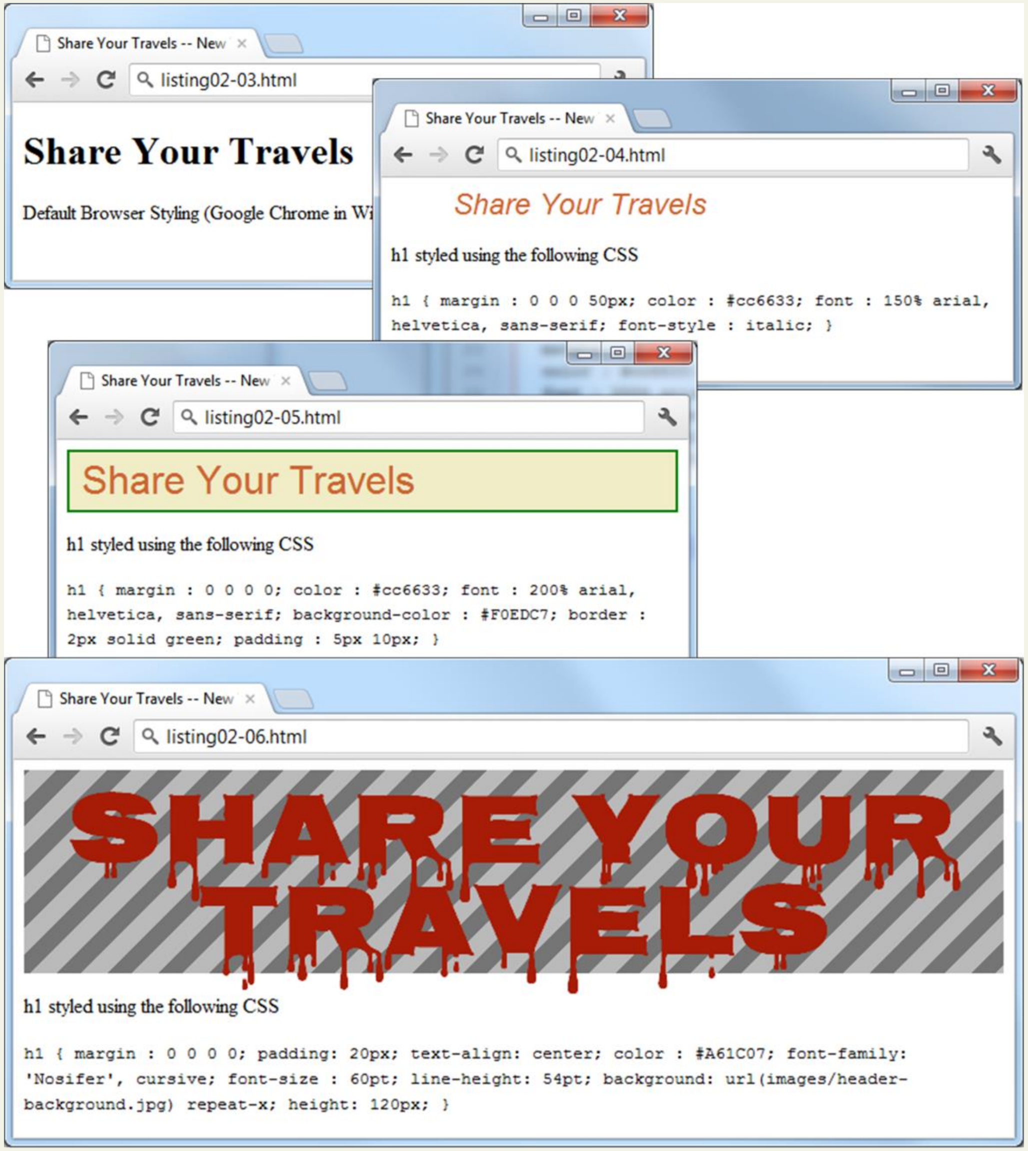
Outline
gsnedders.html5.org/outliner/process.py

1. Term Paper Outline
   1. Introduction
   2. Background
      1. Previous Research
      2. Unresolved Issues
   3. My Solution
      1. Methodology
      2. Results
      3. Discusssion
   4. Conclusion

Headings

The browser has its own default styling for each heading level.

However, these are easily modified and customized via CSS.


Share Your Travels -- New
listing02-03.html
Share Your Travels -- New
listing02-04.html
Share Your Travels -- New
listing02-05.html
Share Your Travels -- New
listing02-06.html

Share Your Travels

Default Browser Styling (Google Chrome in Wi

Share Your Travels

h1 styled using the following CSS

```
h1 { margin : 0 0 0 50px; color : #cc6633; font : 150% arial,
helvetica, sans-serif; font-style : italic; }
```

Share Your Travels

h1 styled using the following CSS

```
h1 { margin : 0 0 0 0; color : #cc6633; font : 200% arial,
helvetica, sans-serif; background-color : #F0EDC7; border :
2px solid green; padding : 5px 10px; }
```

SHARE YOUR TRAVELS

h1 styled using the following CSS

```
h1 { margin : 0 0 0 0; padding: 20px; text-align: center; color : #A61C07; font-family:
'Nosifer', cursive; font-size : 60pt; line-height: 54pt; background: url(images/header-
background.jpg) repeat-x; height: 120px; }
```

Headings

Be semantically accurate

In practice, specify a heading level that is semantically accurate.

Do not choose a heading level because of its default presentation

- e.g., choosing <h3> because you want your text to be bold and 16pt

Rather, choose the heading level because it is appropriate

- e.g., choosing <h3> because it is a third level heading and not a primary or secondary heading

2 Paragraphs and Divisions

<p>

Paragraphs are the most basic unit of text in an HTML document.

Notice that the **<p>** tag is a container and can contain HTML and other **inline HTML elements**

- inline HTML elements refers to HTML elements that do not cause a paragraph break but are part of the regular “flow” of the text.

This **<div>** tag is also a container element and is used to create a logical grouping of content

- The <div> element has no intrinsic presentation and used to mark out sections.

Paragraphs and Divisions

Example

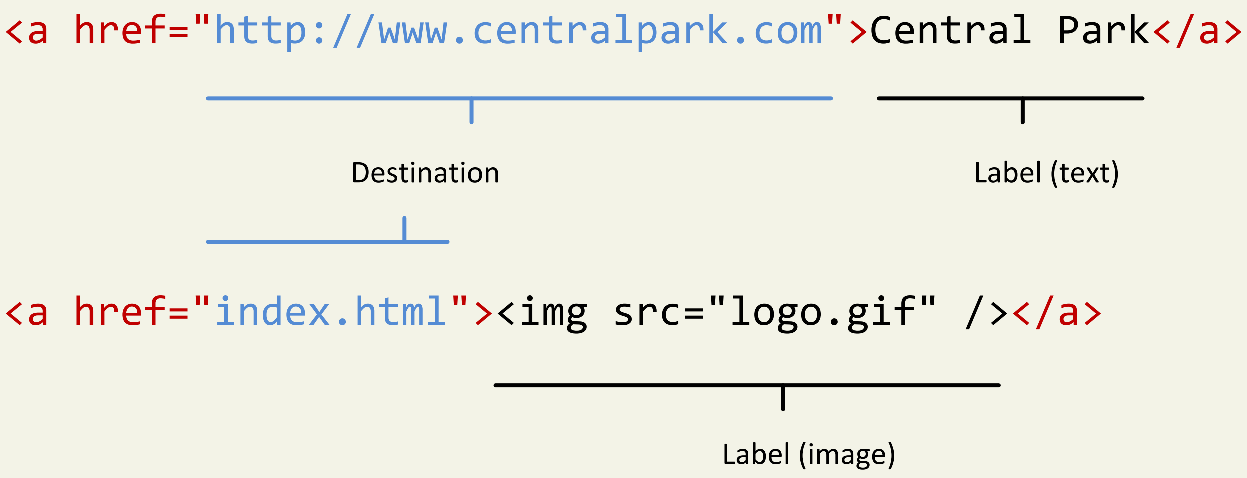
```
<!DOCTYPE html>
<html>
        <title>My Awesome Page</title>
<body>
        <div>
                <p>I’m a paragraph!</p>
        </div>
        <div>
                <h1>This is another div!</h1>
                <p>Some content would go here</p>
        </div>
</body>
</html>
```

❸ Links

<a>

Links are created using the **<a>** element (the “a” stands for anchor).

A link has two main parts: the destination and the label.



```
<a href="http://www.centralpark.com">Central Park</a>
```

Destination

Label (text)

```
<a href="index.html"><img src="logo.gif" /></a>
```

Label (image)

Types of Links

Link to external site

```
<a href="http://www.centralpark.com">Central Park</a>
```

Link to resource on external site

```
<a href="http://www.centralpark.com/logo.gif">Central Park</a>
```

Link to another page on same site as this page

```
<a href="index.html">Home</a>
```

Link to another place on the same page

```
<a href="#top">Go to Top of Document</a>
...
<a name="top">
```

Defines anchor for a link to another place on same page

Types of Links

Link to specific place on another page

```
<a href="productX.html#reviews">Reviews for product X</a>
```

Link to email

```
<a href="mailto:person@somewhere.com">Someone</a>
```

Link to JavaScript function

```
<a href="javascript:OpenAnnoyingPopup();">See This</a>
```

Link to telephone (automatically dials the number
when user clicks on it using a smartphone browser)

```
<a href="tel:+18009220579">Call toll free (800) 922-0579</a>
```

Link Text

Some guidance … or … don't "Click Here"

Links with the label "Click Here" were once a staple of the web.

Today, such links are frowned upon, since:

- they do not tell users where the link will take them
- as a verb "click" is becoming increasingly inaccurate when one takes into account the growth of mobile browsers.

Instead, textual link labels should be descriptive.

~~"Click here to see the race results"~~

"Race Results" or **"See Race Results".**

URL Absolute Referencing

For external resources

When referencing a page or resource on an external site, a full **absolute reference** is required: that is,

- the protocol (typically, http://),
- the domain name,
- any paths, and then finally
- the file name of the desired resource.

URL Relative Referencing

An essential skill

We also need to be able to successfully reference files within your site.

This requires learning the syntax for **relative referencing**.

When referencing a resource that is on the same server as your HTML document, then you can use briefer relative referencing. If the URL does not include the “http://” then the browser will request the current server for the file.

URL Relative Referencing

If all the resources for the site reside within the same **directory** (also referred to as a **folder**), then you can reference those other resources simply via their filename.

However, most real-world sites contain too many files to put them all within a single directory.

For these situations, a relative pathname is required along with the filename.

The **pathname** tells the browser where to locate the file on the server.

Pathnames

Pathnames on the web follow Unix conventions.

- Forward slashes (“/”) are used to separate directory names from each other and from file names.
- Double-periods (“..”) are used to reference a directory “above” the current one in the directory tree.

URL Relative Referencing

Relative Link Type	Example
Same directory	<a href="example.html">
Child Directory	<a href="images/logo.gif">
Grandchild/Descendant Directory	<a href="css/images/background.gif">
Parent/Ancestor Directory	<a href="../about.html"> <a href="../../about.html”>
Sibling Directory	<a href="../images/about.html">
Root Reference	<a href="/images/background.gif">

Inline Text Elements

Do not disrupt the flow

Inline elements do not disrupt the flow of text (i.e., cause a line break).

HTML5 defines over 30 of these elements.

http://www.w3schools.com/html/html_blocks.asp

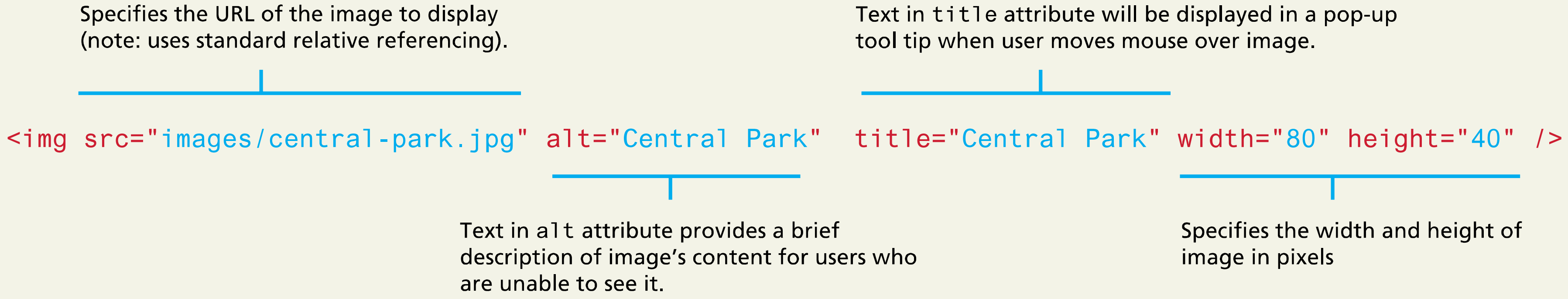
- <a>
- <abbr>
- <br>
- <cite>
- <code>
- <em>
- <mark>
- <small>
- <span>
- <strong>
- <time>

Images

While the **<img>** tag is the oldest method for displaying an image, it is not the only way.

For purely decorative images, it makes semantic sense to keep such images out of the markup and in CSS

When the images are content, then the <img> tag is the semantically appropriate approach.



Specifies the URL of the image to display (note: uses standard relative referencing).

Text in `title` attribute will be displayed in a pop-up tool tip when user moves mouse over image.

```
<img src="images/central-park.jpg" alt="Central Park" title="Central Park" width="80" height="40" />
```

Text in `alt` attribute provides a brief description of image’s content for users who are unable to see it.

Specifies the width and height of image in pixels

Lists

HTML provides three types of lists

Unordered lists.(<ul>) Collections of items in no particular order; these are by default rendered by the browser as a bulleted list.

Ordered lists. (<ol>) Collections of items that have a set order; these are by default rendered by the browser as a numbered list.

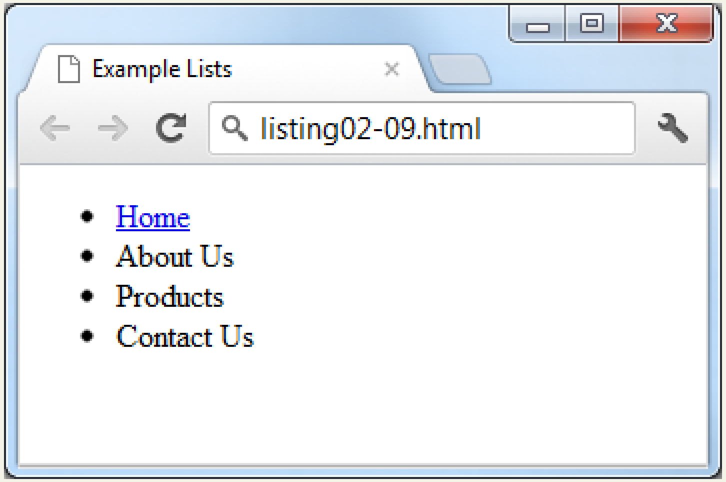
Description lists. <dl> Collection of name and definition pairs. These tend to be used infrequently. Perhaps the most common example would be a FAQ list.

Lists

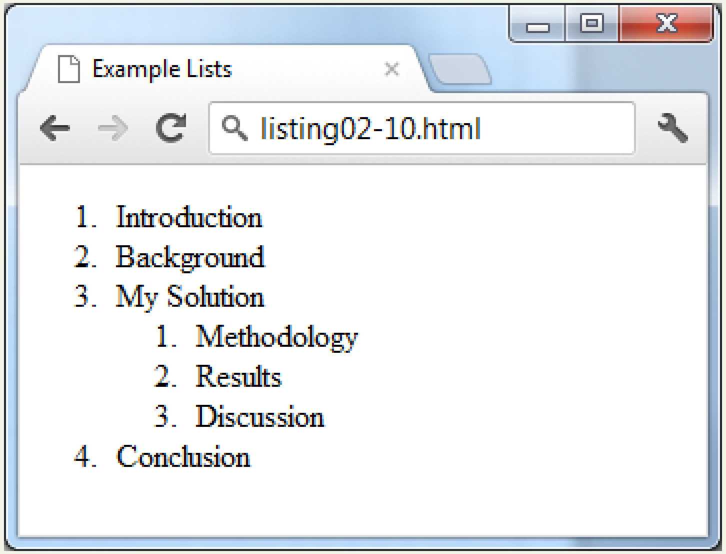
Notice that the list item element can contain other HTML elements

```
<ul>
   <li><a href="index.html">Home</a></li>
   <li>About Us</li>
   <li>Products</li>
   <li>Contact Us</li>
</ul>
```

```
<ol>
   <li>Introduction</li>
   <li>Background</li>
   <li>My Solution</li>
   <li>
     <ol>
        <li>Methodology</li>
        <li>Results</li>
        <li>Discussion</li>
     </ol>
   </li>
   <li>Conclusion</li>
</ol>
```



- Home
- About Us
- Products
- Contact Us



1. Introduction
2. Background
3. My Solution
    1. Methodology
    2. Results
    3. Discussion
4. Conclusion

Character Entities

These are special characters for symbols for which

- There is either no easy way to type them via a keyboard
- Which have a reserved meaning in HTML (like“<“)

Entity	Description
&nbsp;	Nonbreakable space
&lt;	<
&gt;	>
&copy;	©
&trade;	™

Question

Is this a valid link element?

<a href=“google.com”>Go to Google</a>

A. Yes

B. No

Question

Is this a valid link element?

<a href=“google.com”>Go to Google</a>

A. Yes

B. No

Chapter 3

1 What Is HTML and Where Did It Come from?

2 HTML Syntax

3 Semantic Markup

4 Structure of HTML Documents

5 Quick Tour of HTML Elements

6 HTML5 Semantic Structure Elements

7 Summary

HTML5 Semantic Elements

Why are they needed?

One substantial problem with modern, pre-HTML5 semantic markup:

- Most complex web sites are absolutely packed solid with <div> elements.
- <div> elements can make the resulting markup confusing and hard to modify.

Developers typically try to bring some sense and order to the <div> chaos by using id or class names that provide some clue as to their meaning.

Header and Footer

<header> <footer>

Most web site pages have a recognizable header and footer section.

Typically the **header** contains

- the site logo
- title (and perhaps additional subtitles or taglines)
- horizontal navigation links, and
- perhaps one or two horizontal banners.

1

```
<header>
   . . .
   <nav>
      . . .
   </nav>
</header>
```

Header and Footer

<header> <footer>

The typical footer contains less important material, such as

- smaller text versions of the navigation,
- copyright notices,
- information about the site’s privacy policy, and
- perhaps twitter feeds or links to other social sites.

```
<footer>
   . . .
</footer>
```

9

Header and Footer

<header> <footer>

- *A header element is intended to usually contain the section’s heading (an h1– h6 element), but this is not required.*

- *The header element can also be used to wrap a section’s table of contents, a search form, or any relevant logos.*

```
<header>
<img src="logo.gif" alt="logo" />
<h1>Fundamentals of Web
Development</h1>
…
</header>
<article>
<header>
<h2>HTML5 Semantic Structure
Elements</h2>
<p> By <em>Randy
Connolly</em></p>
</header>
…
</article>
```

Navigation

<nav>

The **<nav>** element represents a section of a page that contains links to other pages or to other parts within the same page.

- The browser does not apply any special presentation to the <nav> element.
- The <nav> element was intended to be used for major navigation blocks, presumably the global and secondary navigation systems.
- Improves accessibility

Navigation

<nav>

```
<header>
    <img src="logo.gif" alt="logo" />
    <h1>Fundamentals of Web Development</h1>
    <nav>
        <ul>
            <li><a href="index.html">Home</a></li>
            <li><a href="about.html">About Us</a></li>
            <li><a href="browse.html">Browse</a></li>
        </ul>
    </nav>
</header>
```

Main

<main>

- **<main>** is meant to contain the main unique content of the document.
- <main> provides a semantic replacement for markup such as <div id="main"> or <div id="main-content">

Articles and Sections

<article> <section>

The **<section>** element represents a section of a document, typically with a title or heading.

The **<article>** element represents a section of content that forms an independent part of a document or site;

i.e. a magazine or newspaper article, or a blog entry.

Sections versus Divs

How to decide which to use

- The WHATWG specification warns readers that the <section> element is **not** a generic container element. HTML already has the <div> element for such uses.
- When an element is needed only for styling purposes, use the <div> element instead.
- To decide whether or not to use the <section> element is to ask if it is appropriate for the element's contents to be listed explicitly in the document's outline.
- If so, then use a <section>; otherwise use a <div>.

Figure and Figure Captions

<figure> <figcaption>

The W3C Recommendation I

- <figure> element can be used not just for images but for any type of *essential* content that could be moved to a different location in the page or document and the rest of the document would still make sense.
- Not needed for every image (i.e. logo)

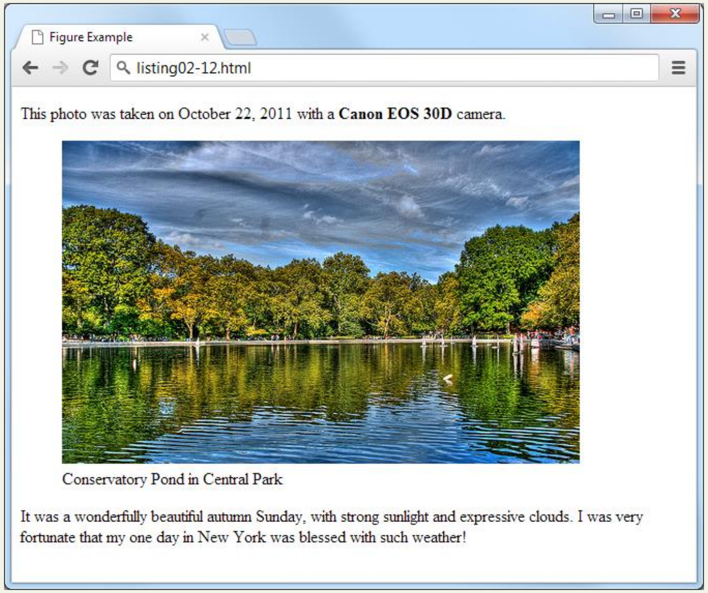
Figure and Figure Captions

Figure could be moved to a different location in document

…

But it has to exist in the document (i.e., the figure isn't optional)

```
<p>This photo was taken on October 22, 2011 with a Canon EOS 30D camera.</p>
<figure>
  <img src="images/central-park.jpg" alt="Central Park" /><br/>
  <figcaption>Conservatory Pond in Central Park</figcaption>
</figure>
<p>
It was a wonderfully beautiful autumn Sunday, with strong sunlight and
expressive clouds. I was very fortunate that my one day in New York was
blessed with such weather!
</p>
```



This photo was taken on October 22, 2011 with a **Canon EOS 30D** camera.

Conservatory Pond in Central Park

It was a wonderfully beautiful autumn Sunday, with strong sunlight and expressive clouds. I was very fortunate that my one day in New York was blessed with such weather!

Aside

<aside>

The **<aside>** element is similar to the **<figure>** element in that it is used for marking up content that is separate from the main content on the page.

<aside> element “represents a section of a page that consists of content that is tangentially related to the content around the aside element.”

The **<aside>** element could thus be used for sidebars, pull quotes, groups of advertising images, or any other grouping of non-essential elements.

Chapter 3

1 What Is HTML and Where Did It Come from?

2 HTML Syntax

3 Semantic Markup

4 Structure of HTML Documents

5 Quick Tour of HTML Elements

6 HTML5 Semantic Structure Elements

7 Summary

Up Next

- Review and know basic HTML5 tags
- Chapter 4 - CSS