

HTML and CSS

Week 2 Lecture

**COMMONWEALTH OF
Copyright Regulations 1969
WARNING**

This material has been reproduced and communicated to you by or on behalf of the University of Sydney pursuant to Part VB of the Copyright Act 1968 (**the Act**).

The material in this communication may be subject to copyright under the Act. Any further reproduction or communication of this material by you may be the subject of copyright protection under the Act.

Do not remove this notice.

Most materials and diagrams in this slides are from chapter 2 and 3 of Fundamentals of Web Development

Outline

- **HTML**
 - **Syntax**
 - **Structure**
 - **Quick Tour**
 - **Semantic Markups**
- **CSS**
 - **Syntax and Location**
 - **Selector**
 - **The box model**
 - **Text Styling**

HTML

- “To publish information for global distribution, one needs a universally understood language, a kind of publishing mother tongue that all computers may potentially understand. The publishing language used by the World Wide Web is HTML (from Hyper Text Markup Language”
- A markup language is simply a way of annotating a document in such a way to make the annotations distinct from the text being annotated.
 - E.g. Latex
- HTML is used world wide, standardization is essential
 - The W3C is the main standards organization for the World Wide Web.

Key events in the history of HTML

- First public specification proposed in 1991 by Tim Berners-Lee
- Was adopted and standardized by World Wide Web Consortium (W3C) in 1997
- The HTML specification was frozen by W3C in 1998 at version 4.01
- W3C also spent efforts on developing XHTML (XML + HTML) in late 1990s
 - Survived till XHTML 2.0 in mid 2000
- In mid 2000, a small working group WHATWG (Web Hypertext Application Technology Working Group) under W3C was formed by developers at Opera and Mozilla to challenge the idea of XML+HTML
 - More practical approach, e.g. instead of forcing all browsers and developers to support a standard set of markups, it would specify how should browser deal with invalid markup.
- In 2009, W3C stopped work on XHTML 2.0 and adopted the work by WHATWG and named it **HTML5**

HTML5

- 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.

HTML: Elements and Attributes

- **HTML documents** are composed of textual content and HTML elements.
- An **HTML element** can contain text, other elements, or be empty. It is identified in the HTML document by tags.
- HTML elements can also contain attributes. An **HTML attribute** is a **name=value** pair that provides more information about the HTML element.

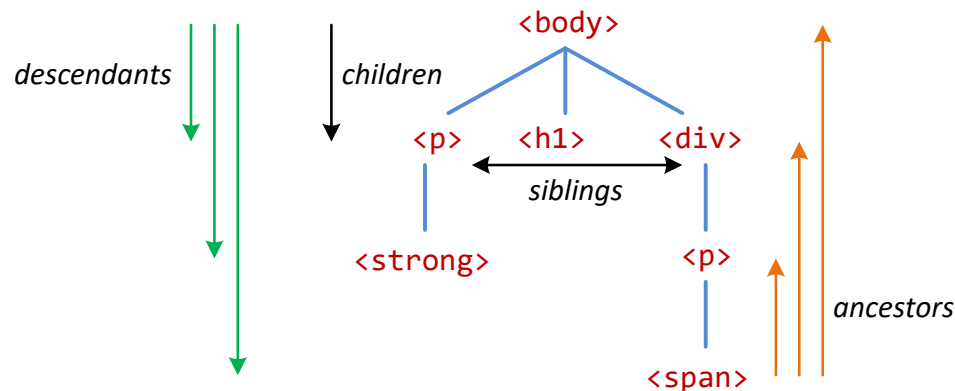
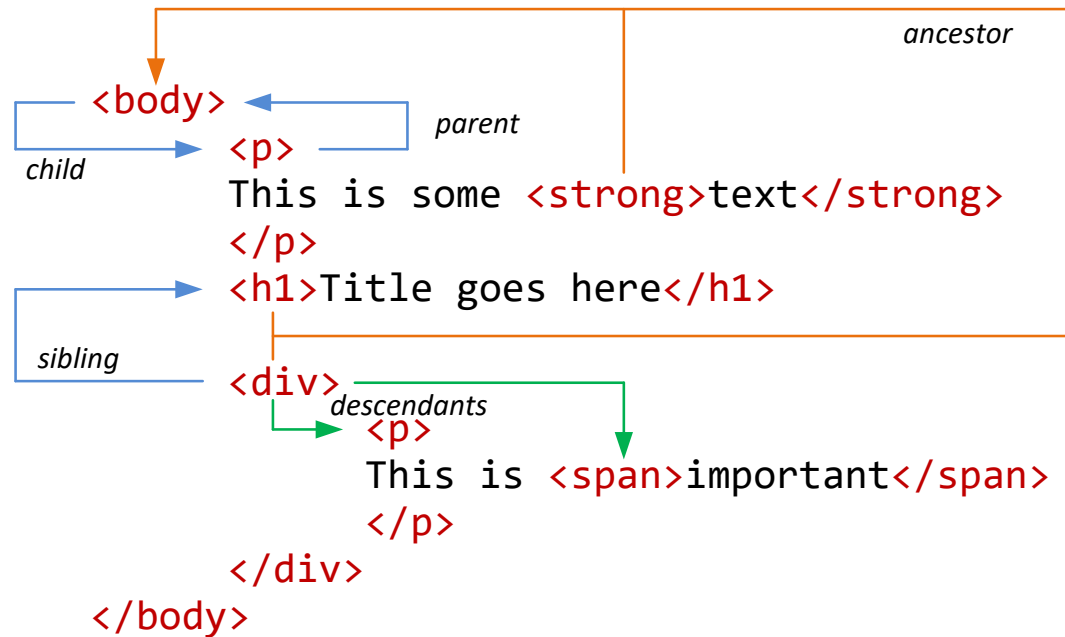
What HTML lets you do

- Insert images using the **** tag
- Create links with the **<a>** tag
- Create lists with the ****, **** and **** tags
- Create headings with **<H1>**, **<H2>**, ..., **<H6>**
- Define metadata with **<meta>** tag
- And much more...

Elements and Attributes



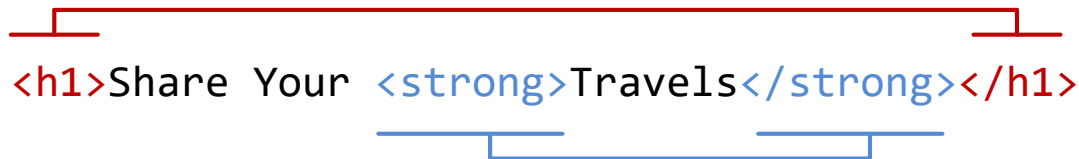
Nesting HTML elements



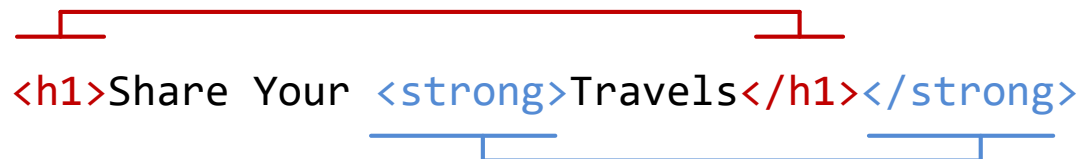
Nesting HTML elements

- 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



The diagram illustrates correct HTML nesting. A red bracket above the text spans from the start of the opening `<h1>` tag to the end of the closing `</h1>` tag. Below the text, a blue bracket spans from the start of the opening `` tag to the end of the closing `` tag. The text is: `<h1>Share Your Travels</h1>`.



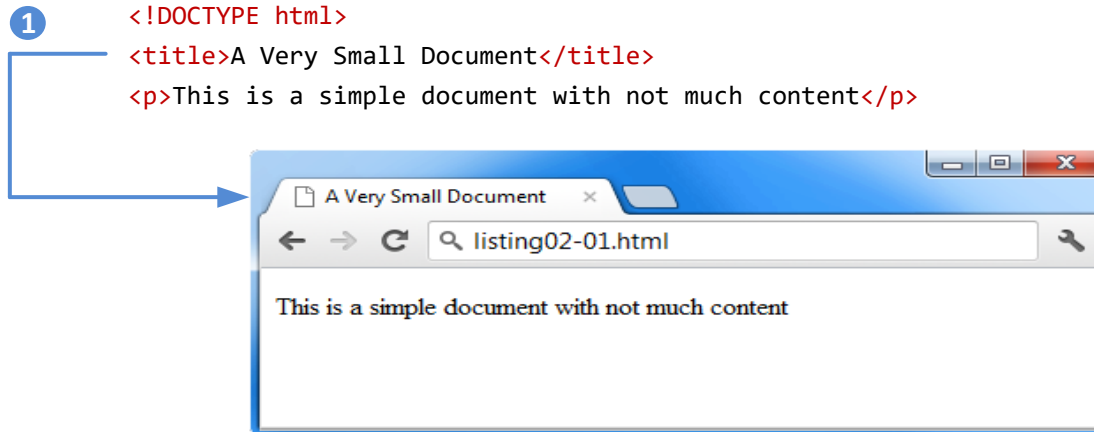
The diagram illustrates incorrect HTML nesting. A red bracket above the text spans from the start of the opening `<h1>` tag to the end of the closing `</h1>` tag. Below the text, a blue bracket spans from the start of the opening `` tag to the end of the closing `` tag. The text is: `<h1>Share Your Travels</h1>`. The closing `</h1>` tag occurs before the closing `` tag, which is visually incorrect as the blue bracket extends further to the right than the red bracket.

Incorrect Nesting

Outline

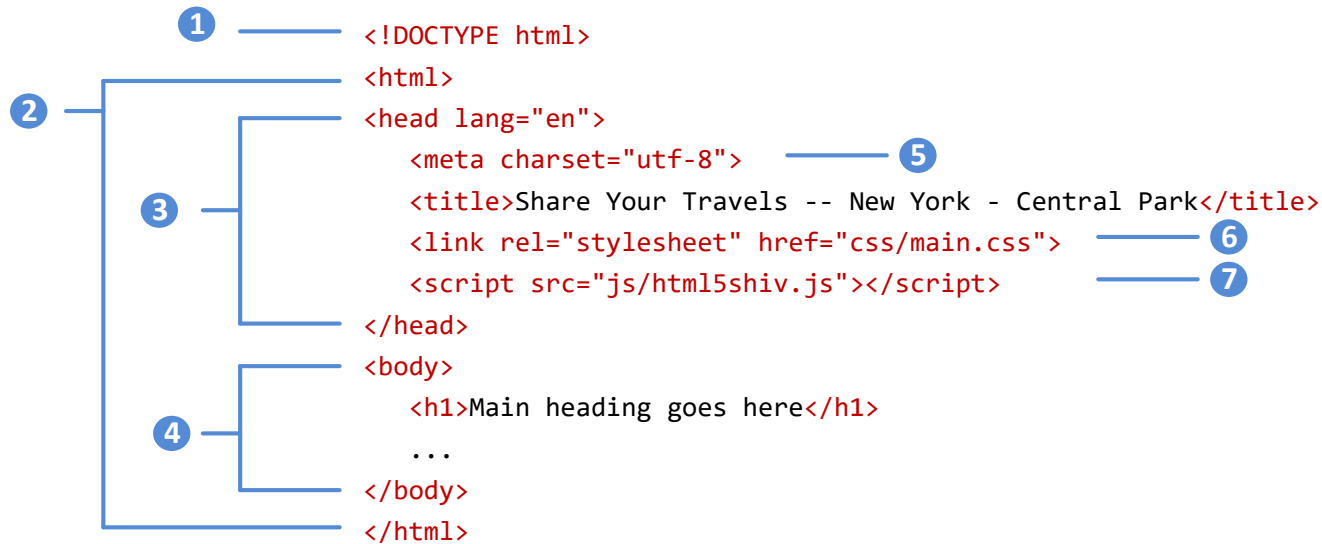
- **HTML**
 - Syntax
 - **Structure**
 - Quick Tour
 - Semantic Markups
- **CSS**
 - Syntax and Location
 - **Selector**
 - The box model
 - Text Styling

Simplest HTML document



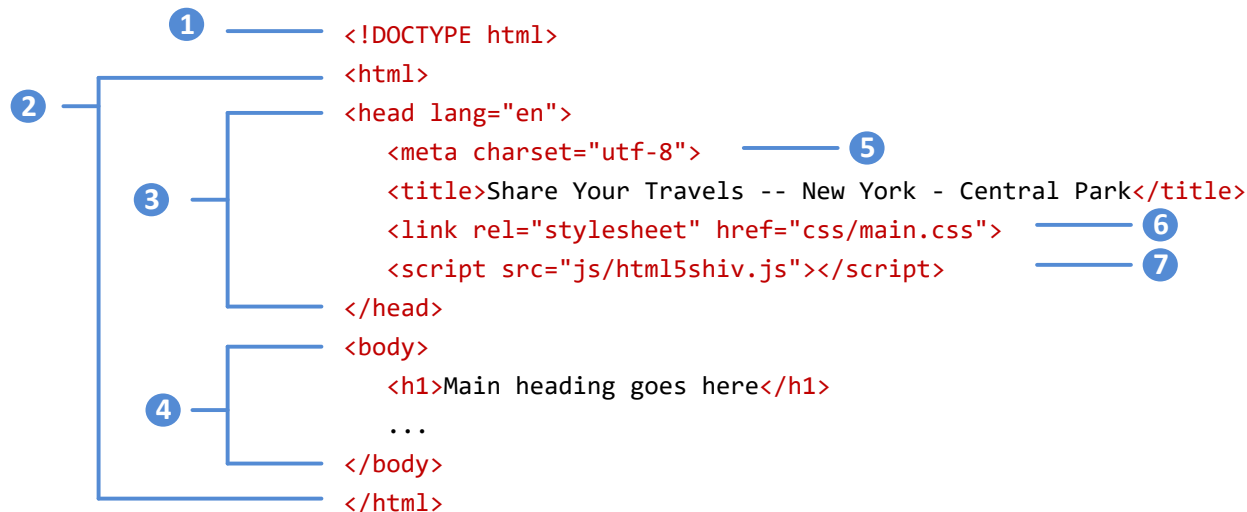
The `<title>` element (Item 1) is used to provide a broad description of the content. The title is not displayed within the browser window. Instead, the title is typically displayed by the browser in its window and/or tab.

A more complete document



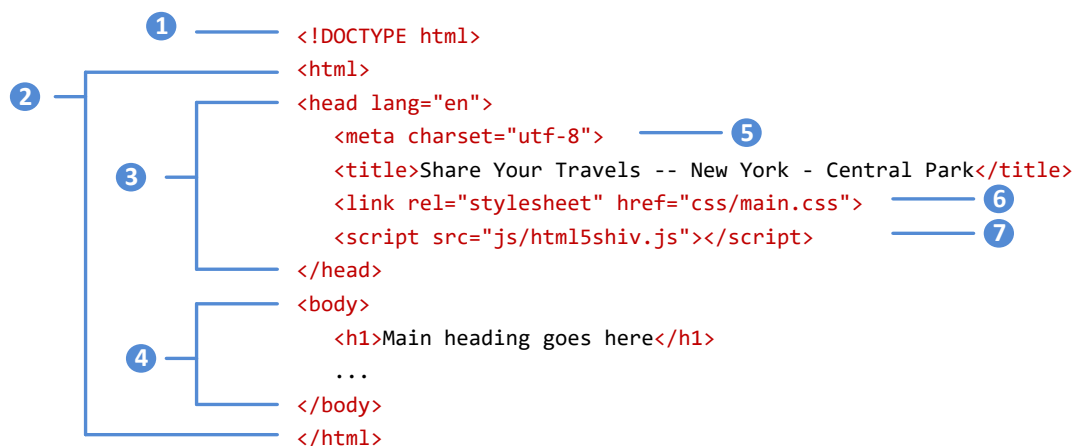
DOCTYPE

- 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.



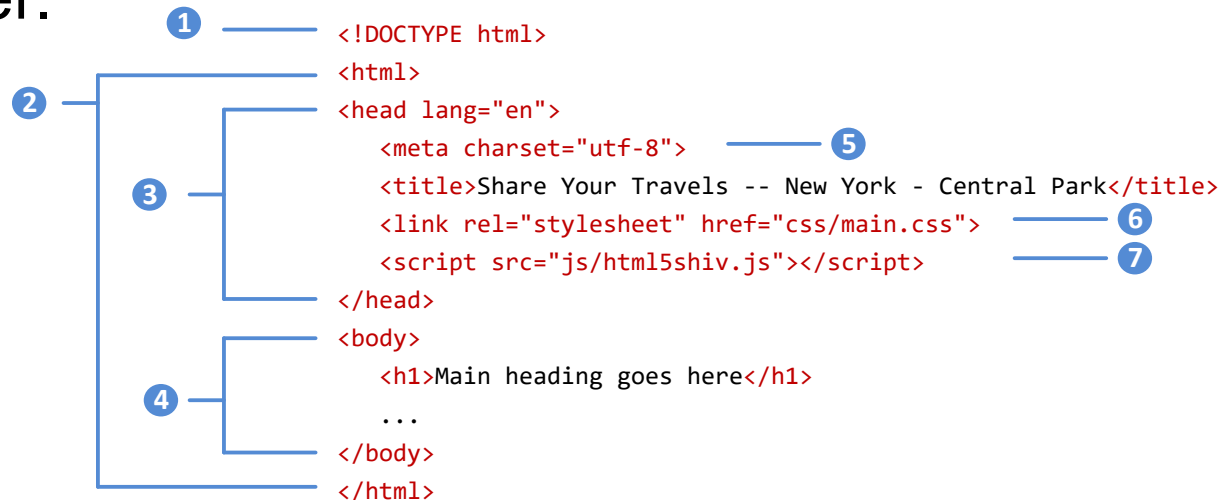
HTML, Head, and Body

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



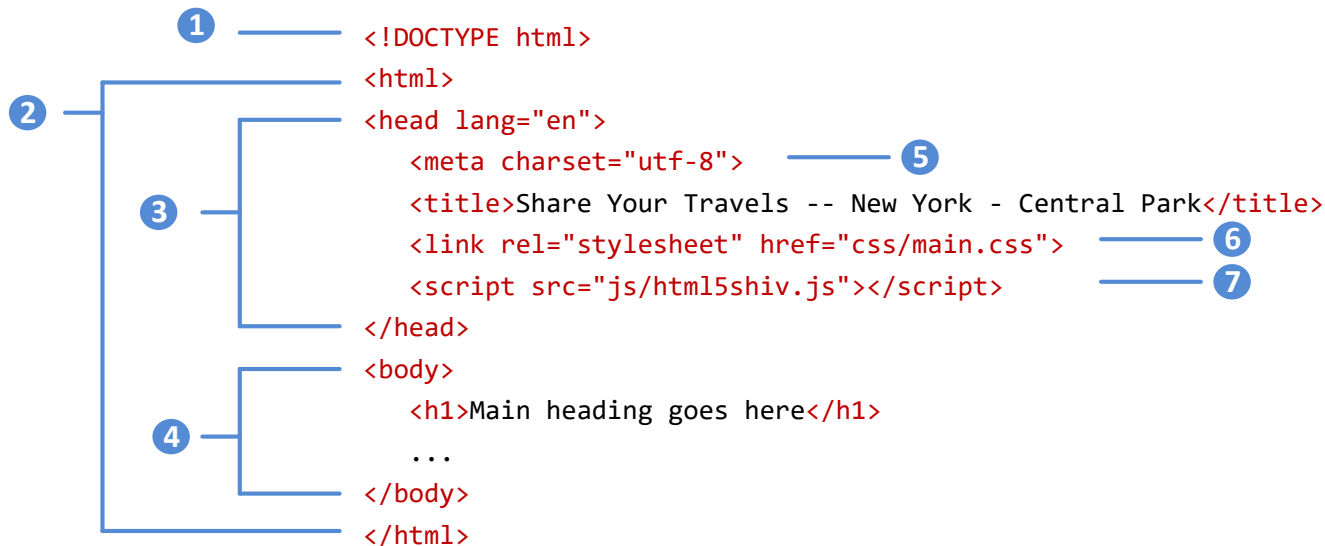
Head and Body

- 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.



Inside the head

- 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.



Outline

- **HTML**
 - Syntax
 - Structure
 - **Quick Tour**
 - Semantic Markups
- **CSS**
 - Syntax and Location
 - **Selector**
 - The box model
 - Text Styling

Sample Document

<body>

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

2 `<h2>New York - Central Park</h2>`

`<p>Photo by Randy Connolly</p>`

`<p>This photo of Conservatory Pond in`

`Central Park` 3

`New York City was taken on October 22, 2011 with a`

`Canon EOS 30D camera.`

`</p>` 4

5 ``

`<h3>Reviews</h3>`

6 `<div>` 7

`<p>By Ricardo on <time>September 15, 2012</time></p>`

`<p>Easy on the HDR buddy.</p>`

`</div>`

`<div>`

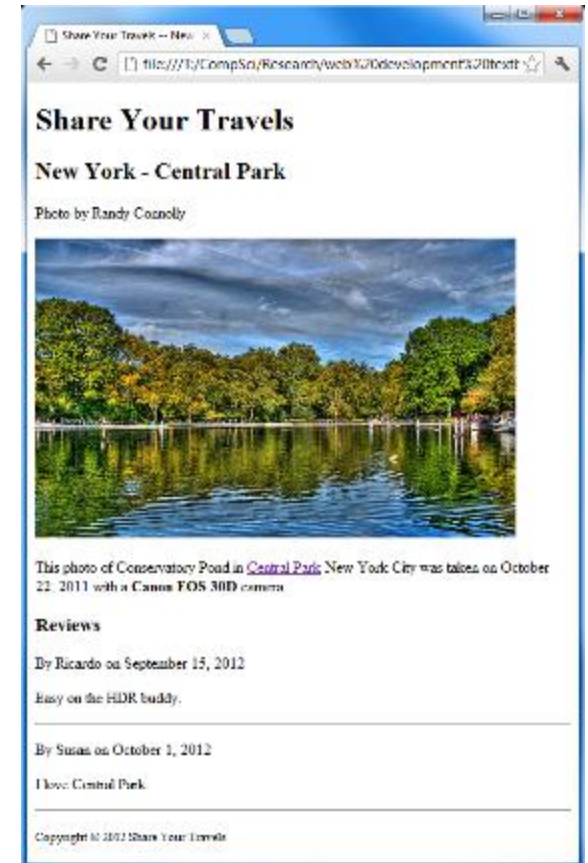
`<p>By Susan on <time>October 1, 2012</time></p>`

`<p>I love Central Park.</p>`

`</div>` 8

`<p><small>Copyright © 2012 Share Your Travels</small></p>`

`</body>` 9



Headings

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.

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

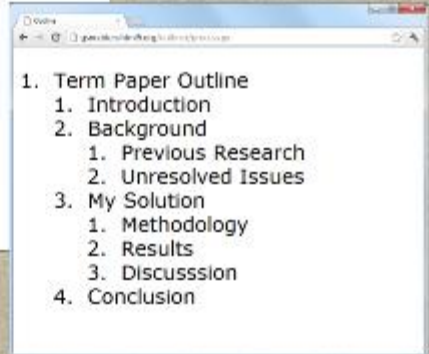
```
<!DOCTYPE html>
<html>
<head lang="en">
  <meta charset="utf-8">
  <title>Term Paper Outline</title>
</head>
<body>
  <h1>Term Paper Outline</h1>

  <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>Discussion</h3>

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

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

Links

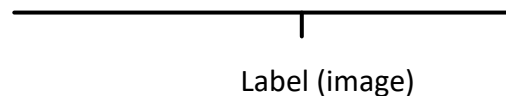
Links are created using the `<a>` element (the “a” stands for anchor).

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

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



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



Types of Links

You can use the anchor element to create a wide range of links:

- Links to external sites (or to individual resources such as images or movies on an external site).
- Links to other pages or resources within the current site.
- Links to other places within the current page.
- Links to particular locations on another page.
- Links that are instructions to the browser to start the user's email program.
- Links that are instructions to the browser to execute a Javascript function.

Different link destinations

Link to external site

[Central Park](http://www.centralpark.com)

Link to resource on external site

[Central Park](http://www.centralpark.com/logo.gif)

Link to another page on same site as this page

[Home](index.html)

Link to another place on the same page

[Go to Top of Document](#top)

Link to specific place on another page

[Reviews for product X](productX.html#reviews)

Link to email

[Someone](mailto://person@somewhere.com)

Link to javascript function

[See This](javascript://OpenAnnoyingPopup();)

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

[Call toll free \(800\) 922-0579](tel:+18009220579)

Images

- There are two different ways of specifying images
 - For purely decorative images, such as background gradients and patterns, logos, border art, and so on, it makes semantic sense to keep such images out of the markup and in CSS where they more rightly belong.
 - when the images are content, such as in the images in a gallery or the image of a product in a product details page, then the `` 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 popup
tool tip when user moves mouse over image.

```

```

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

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

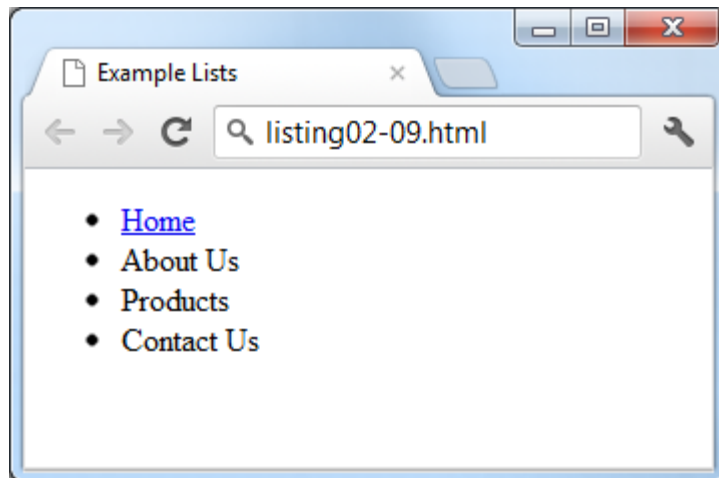
Ordered lists. Collections of items that have a set order; these are by default rendered by the browser as a numbered list.

Definition lists. Collection of name and definition pairs.

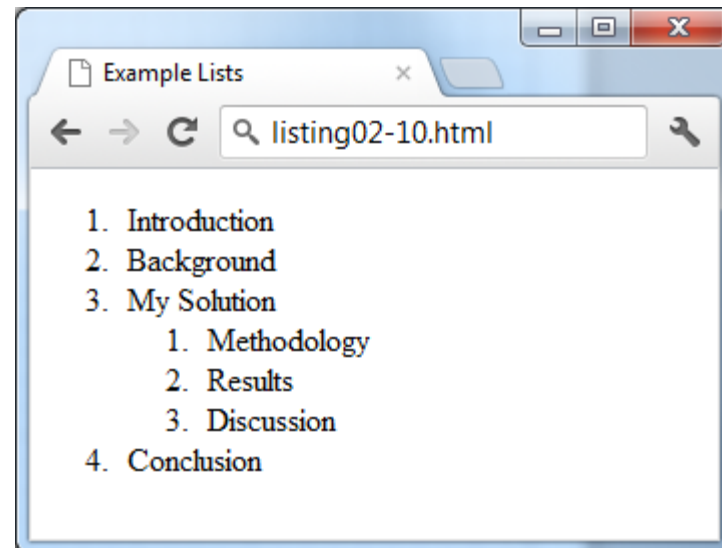
Lists

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>
```



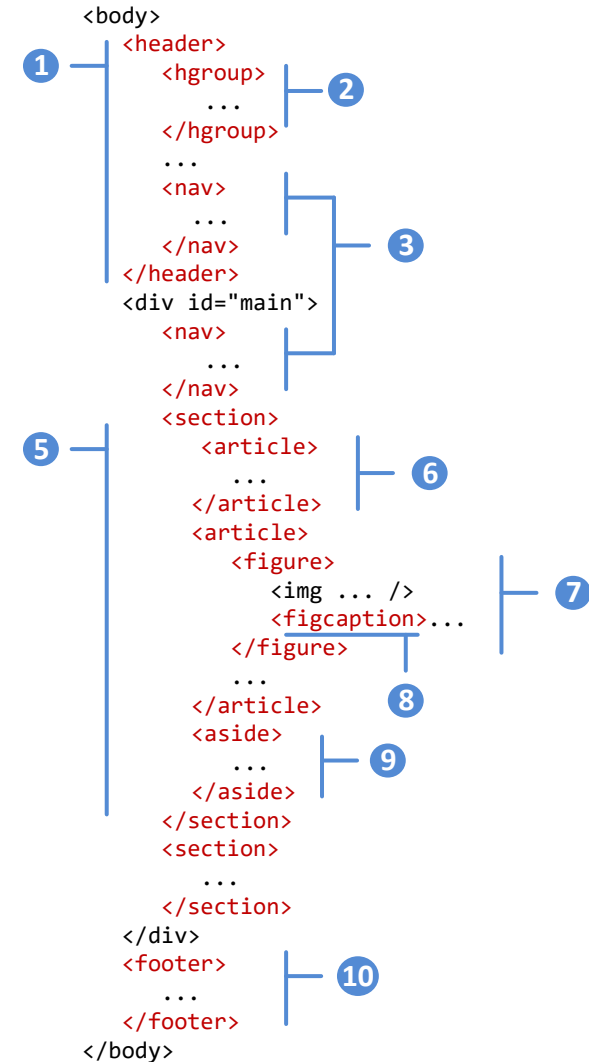
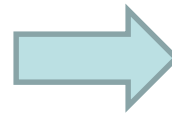
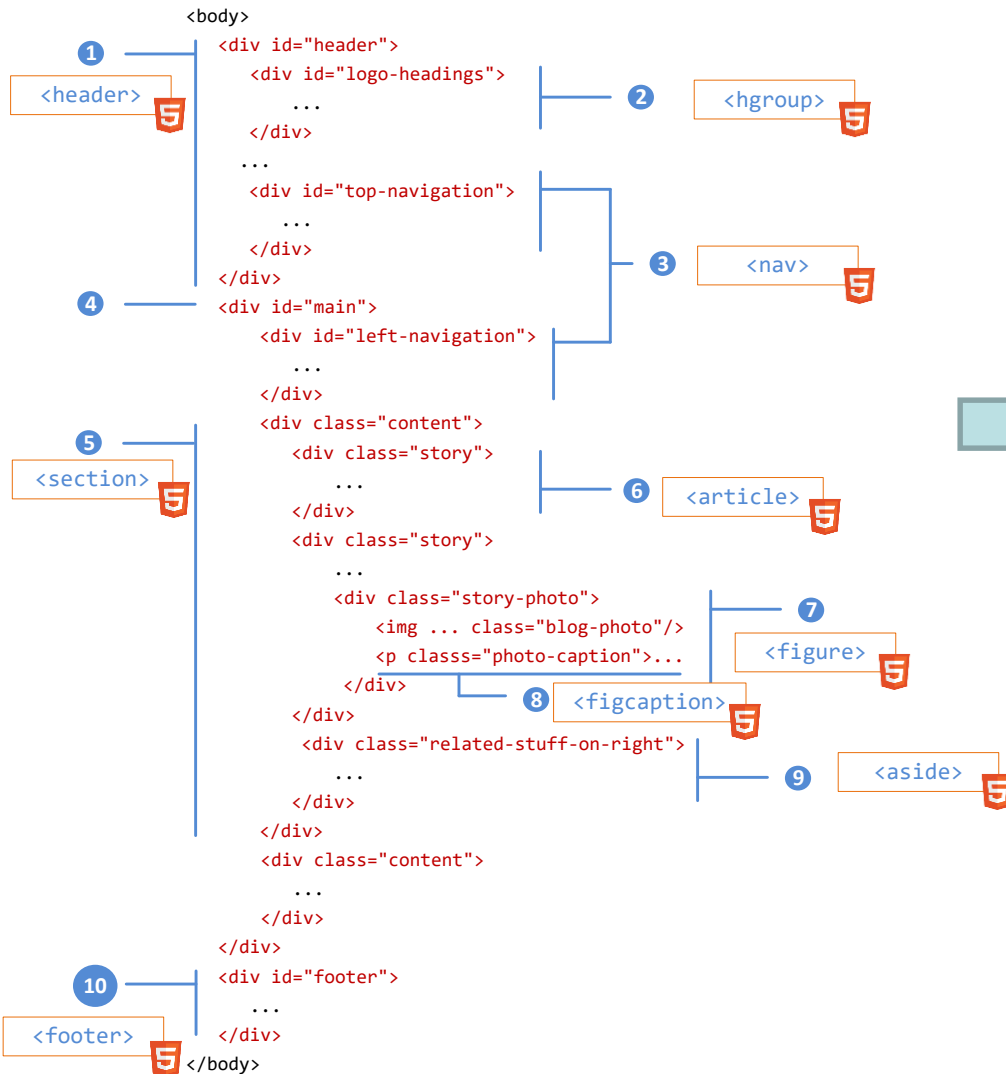
Outline

- **HTML**
 - Syntax
 - Structure
 - Quick Tour
 - **Semantic Markups**
- **CSS**
 - Syntax and Location
 - **Selector**
 - The box model
 - Text Styling

Semantic markups

- The commonly agreed good practice
 - HTML should only focus on the structure of a document
 - How the content should be displayed in a browser window is best left to CSS
- HTML5 has introduced many new semantic tags
 - `<article>`, `<section>`, `<header>`, `<footer>`, `<aside>`, etc to replace tags like
 - `<div id="header">`, `<div id = "footer">`, etc

XHTML versus HTML5



Header and Footer

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

Navigation

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

Like the other new HTML5 semantic elements, the browser does not apply any special presentation to the **<nav>** element.

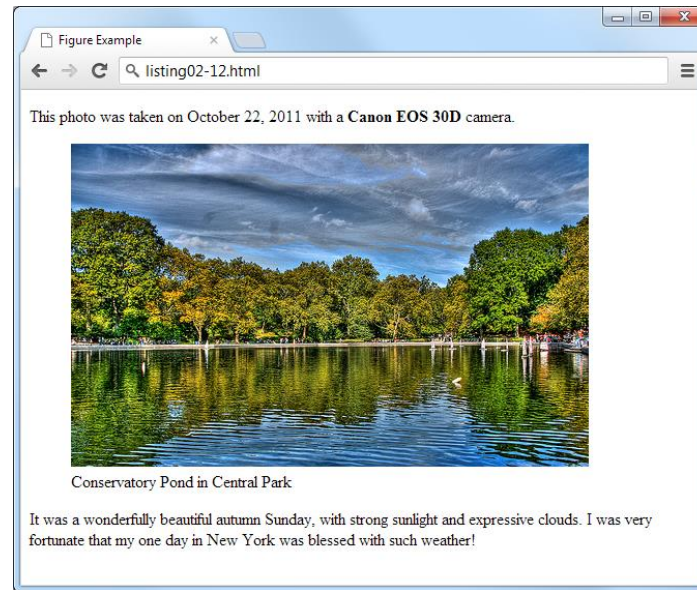
The **<nav>** element was intended to be used for major navigation blocks, presumably the global and secondary navigation systems.

```
<header>
  
  <h1>Fundamentals of Web Development</h1>
  <nav role="navigation">
    <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>
```

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>  
  <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>
```



Outline

- **HTML**
 - Syntax
 - Structure
 - Quick Tour
 - Semantic Markups
- **CSS**
 - **Syntax and Location**
 - Selector
 - The box model
 - Text Styling

Cascading Style Sheets (CSS)

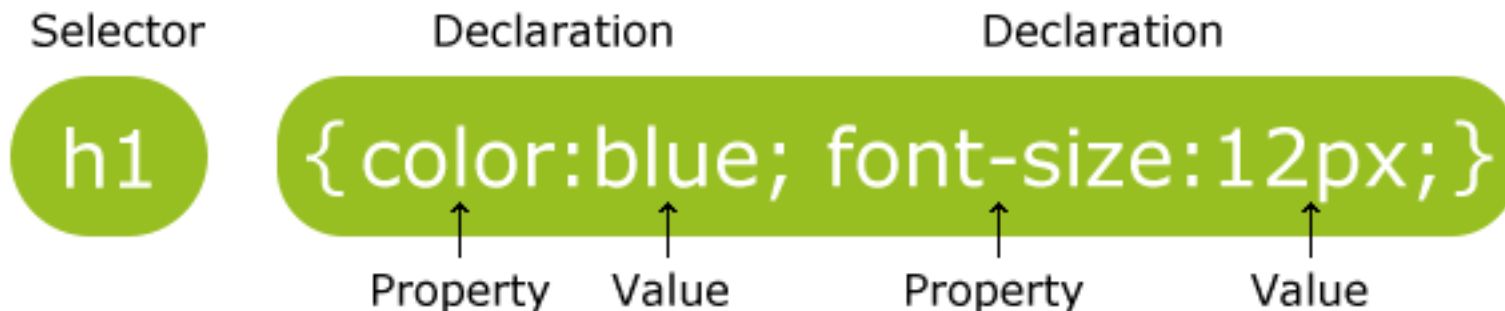
- CSS is a W3C standard for describing the **presentation (or appearance)** of HTML elements.
- With CSS, we can assign
 - font properties,
 - colors,
 - sizes,
 - borders,
 - background images,
 - even the position of elements.

A brief history

- CSS came not too long after the proposal of HTML
- The first recommendation CSS Level 1 (CSS1) was released by W3C in 1996.
- The CSS Level 2 (CSS2) was published in 1998, with a updated version CSS2.1 became official W3C recommendation in 2011
- The current version is CSS3 with a variety of modules at different stages: recommendation, draft and so on
 - backwards-compatible with earlier versions of CSS.
- Browser support has long been an issue of CSS
 - all browsers have left certain parts of the CSS2 Recommendation unimplemented

CSS Syntax

- A CSS document consists of one or more style rules
- Each rule consists of a selector followed by list of declarations (property value pairs)



CSS Syntax

- **Selector** is used to identify the element or elements that will be affected by the styles.
 - Selector can be based on name, class, id, attribute and more
- The **property** name are predefined by CSS standard
- The **property value** are either predefined keywords such as color name or other values such as length measurement, percentage and so on.
 - The unit of any given value is dependent upon the property

Units of Measurement

- Relative vs. Absolute measurements
 - Relative units are based on the value of something else, such as the size of a parent element.
 - Absolute units have real world size

Unit	Description	Type
in	Inches	Absolute
cm	Centimeters	Absolute
mm	Millimeters	Absolute
pt	Points (equal to 1/72 of an inch)	Absolute
pc	Pica (equal to 1/6 of an inch)	Absolute

Relative Units

Unit	Description	Type
px	Pixel. In CSS2 this is a relative measure, while in CSS3 it is absolute (1/96 of an inch).	Relative (CSS2) Absolute (CSS3)
em	Equal to the computed value of the font-size property of the element on which it is used. When used for font sizes, the em unit is in relation to the font size of the parent.	Relative
%	A measure that is always relative to another value. The precise meaning of % varies depending upon which property it is being used.	Relative
ex	A rarely used relative measure that expresses size in relation to the x-height of an element's font.	Relative
ch	Another rarely used relative measure; this one expresses size in relation to the width of the zero ("0") character of an element's font.	Relative (CSS3 only)
rem	Stands for root em, which is the font size of the root element. Unlike em , which may be different for each element, the rem is constant throughout the document.	Relative (CSS3 only)
vw, vh	Stands for viewport width and viewport height. Both are percentage values (between 0 and 100) of the viewport (browser window). This allows an item to change size when the viewport is resized.	Relative (CSS3 only)

CSS location

- CSS rules can be defined inline with an element, embedded in a HTML document or as external document.
 - `<h1 style = "font-family:arial; color:red">Hello World</h1>`
- Embedded style consists of a list of rules enclosed as an element of `<style>` inside HTML's `<head>` element
- External style are placed in a separate document `xxx.css` and are linked to the HTML file
 - `<link href="style.css" rel="stylesheet">`
- Browser has a default set of rules for pages/elements that do not have a customized style
 - Browser Style Sheet
 - User Style Sheet

What is “Cascading”?

- Cascading is the principle to resolve conflicting style rules
 - Inheritance
 - Specificity
 - Location
- Inheritance
 - Certain properties defined for the <body> element will be inherited by all elements embedded in it
- Specificity
 - Style rules with more specific selector will override rules with more general selector
 - E.g. If both <body> and <p> elements define a font rule, the text inside the <p> element will use the font defined in the <p> element
- Location
 - If inheritance and specificity cannot determine style precedence, the latest (closest) rule wins
 - E.g. inline rule is closer to that element than the rules in an external file

Outline

- **HTML**
 - Syntax
 - Structure
 - Quick Tour
 - Semantic Markups
- **CSS**
 - Syntax and Location
 - **Selector**
 - The box model
 - Text Styling

Selectors

- When defining CSS rules, you will need to first need to use a **selector** to tell the browser which elements will be affected.
- CSS selectors allow you to select
 - individual elements
 - multiple HTML elements,
 - elements that belong together in some way, or
 - elements that are positioned in specific ways in the document hierarchy.

Element Selectors

Uses the HTML element name.

You can select all elements by using the **universal element selector**, which is the * (asterisk) character.

declaration

selector { property: value; property2: value2; } — rule

declaration block

selector

em { color: red; }

property value

p {
margin: 5px 0 10px 0;
font-weight: bold;
font-family: Arial, Helvetica, sans-serif;
}

Grouped Selectors

```
/* commas allow you to group selectors */  
p, div, aside {  
    margin: 0;  
    padding: 0;  
}  
/* the above single grouped selector is equivalent to the  
   following: */  
p {  
    margin: 0;  
    padding: 0;  
}  
div {  
    margin: 0;  
    padding: 0;  
}  
aside {  
    margin: 0;  
    padding: 0;  
}
```

LISTING 3.4 Sample grouped selector

Class Selectors

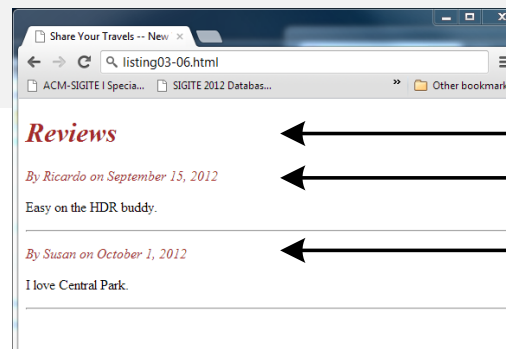
A **class selector** allows you to simultaneously target different HTML elements regardless of their position in the document tree.

If a series of HTML element have been labeled with ***the same class attribute value***, then you can target them for styling by using a class selector, which takes the form: period (.) followed by the class name.

Class Selectors

```
<head>
  <title>Share Your Travels </title>
  <style>
    .first {
      font-style: italic;
      color: brown;
    }
  </style>
</head>
<body>
  <h1 class="first">Reviews</h1>
  <div>
    <p class="first">By Ricardo on <time>September 15, 2012</time></p>
    <p>Easy on the HDR buddy.</p>
  </div>
  <hr/>

  <div>
    <p class="first">By Susan on <time>October 1, 2012</time></p>
    <p>I love Central Park.</p>
  </div>
  <hr/>
</body>
```



```
.first {
  font-style: italic;
  color: brown;
}
```

Id Selectors

An **id selector** allows you to target a specific element by its id attribute regardless of its type or position.

If an HTML element has been labeled with an id attribute, then you can target it for styling by using an id selector, which takes the form: pound/hash (#) followed by the id name.

Id Selectors

```
<head lang="en">
  <meta charset="utf-8">
  <title>Share Your Travels -- New York - Central Park</title>
  <style>
    #latestComment {
      font-style: italic;
      color: brown;
    }
  </style>
</head>
<body>
  <h1>Reviews</h1>
  <div id="latestComment">
    <p>By Ricardo on <time>September 15, 2012</time></p>
    <p>Easy on the HDR buddy.</p>
  </div>
  <hr/>

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



```
#latestComment {
  font-style: italic;
  color: brown;
}
```

Id versus Class Selectors

Id selectors should only be used when referencing a single HTML element since an id attribute can only be assigned to a single HTML element.

Class selectors should be used when (potentially) referencing several related elements.

Pseudo Selectors

A **pseudo-element selector** is a way to select something that does not exist explicitly as an element in the HTML document tree but which is still a recognizable selectable object.

A **pseudo-class selector** does apply to an HTML element, but targets either a particular state or, in CSS3, a variety of family relationships.

The most common use of this type of selectors is for targeting link states.

Pseudo Selectors

```
<head>
  <title>Share Your Travels</title>
  <style>
    a:link {
      text-decoration: underline;
      color: blue;
    }
    a:visited {
      text-decoration: underline;
      color: purple;
    }
    a:hover {
      text-decoration: none;
      font-weight: bold;
    }
    a:active {
      background-color: yellow;
    }
  </style>
</head>
<body>
  <p>Links are an important part of any web page. To learn more about
    links visit the <a href="#">W3C</a> website.</p>
  <nav>
    <ul>
      <li><a href="#">Canada</a></li>
      <li><a href="#">Germany</a></li>
      <li><a href="#">United States</a></li>
    </ul>
  </nav>
</body>
```

LISTING 3.8 Styling a link using pseudo-class selectors

Contextual Selectors

A **contextual selector** (in CSS3 also called **combinators**) allows you to select elements based on their ancestors, descendants, or siblings.

That is, it selects elements based on their context or their relation to other elements in the document tree.

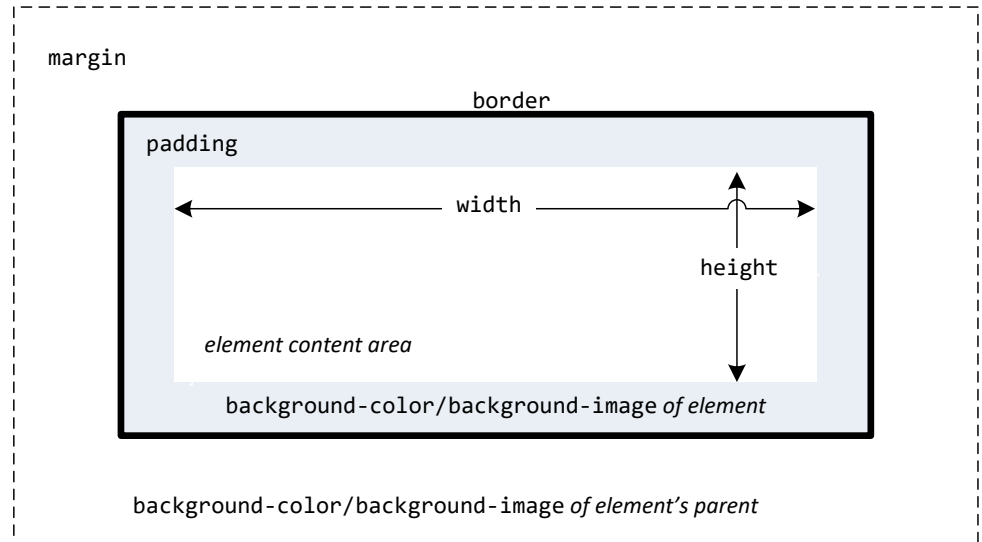
Selector	Matches	Example
Descendant	A specified element that is contained somewhere within another specified element	<div>p Selects a <p> element that is contained somewhere within a <div> element. That is, the <p> can be any descendant, not just a child.</div>
Child	A specified element that is a direct child of the specified element	<div>>h2 Selects an <h2> element that is a child of a <div> element.</div>
Adjacent Sibling	A specified element that is the next sibling (i.e., comes directly after) of the specified element.	<h3>+p Selects the first <p> after any <h3>.</h3>
General Sibling	A specified element that shares the same parent as the specified element.	<h3>~p Selects all the <p> elements that share the same parent as the <h3>.</h3>

Outline

- **HTML**
 - Syntax
 - Structure
 - Quick Tour
 - Semantic Markups
- **CSS**
 - Syntax and Location
 - Selector
 - **The box model**
 - Text Styling

The Box Model

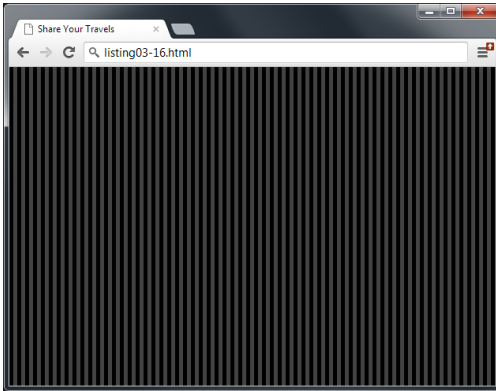
In CSS, all HTML elements exist within an **element box**.



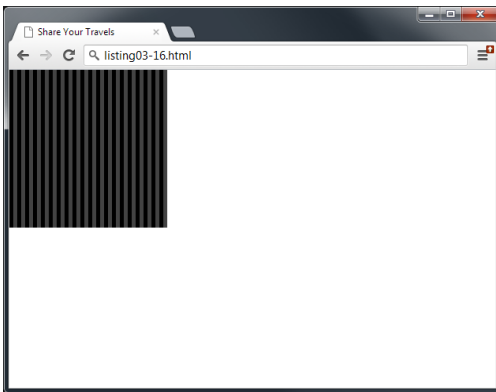
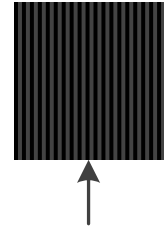
Every CSS rule begins with a selector. The selector identifies which element or elements in the HTML document will be affected by the declarations in the rule. Another way of thinking of selectors is that they are a pattern which is used by the browser to select the HTML elements that will receive

Background properties

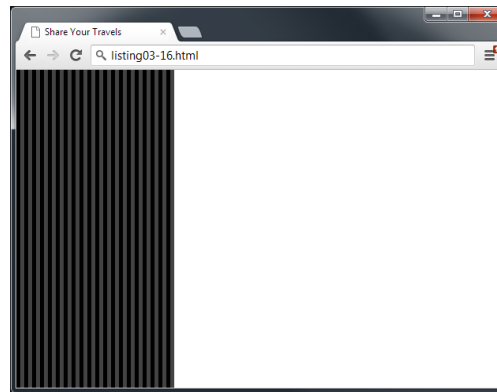
The background color or image fills an element out to its border (if it has one that is).



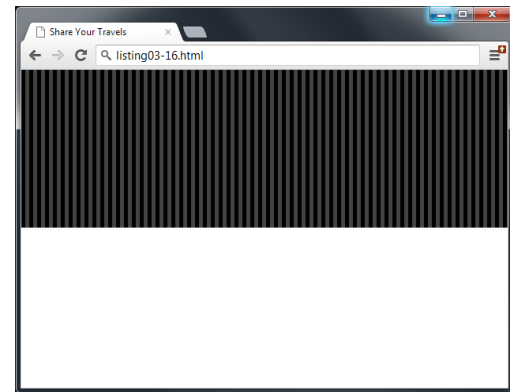
`background-image: url(../images/backgrounds/body-background-tile.gif);`
`background-repeat: repeat;`



`background-repeat: no-repeat;`

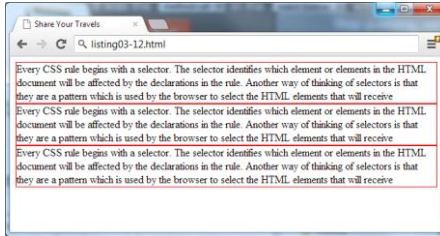


`background-repeat: repeat-y;`

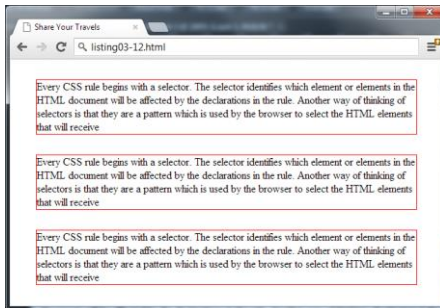


`background-repeat: repeat-x;`

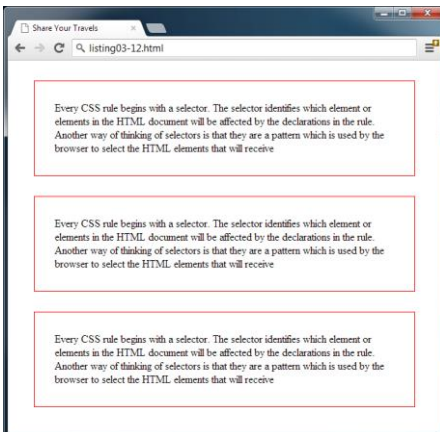
Margins and Padding



```
p {  
  border: solid 1pt red;  
  margin: 0;  
  padding: 0;  
}
```



```
p {  
  border: solid 1pt red;  
  margin: 30px;  
  padding: 0;  
}
```



```
p {  
  border: solid 1pt red;  
  margin: 30px;  
  padding: 30px;  
}
```


Collapsing Margins

When the **vertical** margins of two elements touch,

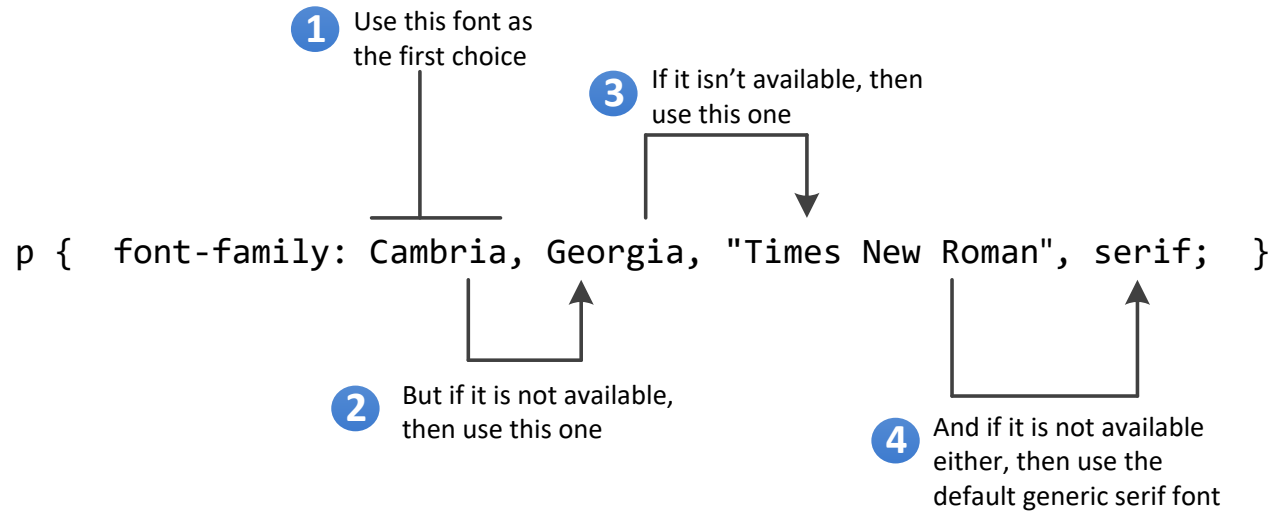
- the largest margin value of the elements will be displayed
- the smaller margin value will be collapsed to zero.

Horizontal margins, on the other hand, **never** collapse.

Outline

- **HTML**
 - Syntax
 - Structure
 - Quick Tour
 - Semantic Markups
- **CSS**
 - Syntax and Location
 - Selector
 - The box model
 - **Text Styling**

Specifying the Font-Family



Font sizes

- How do we specify font sizes?
 - Typically in word document, we specify 10 points, 12 points
 - Printer based, the size of point varied through history
- On screen based media
 - Pixel has been a relatively stable measure
 - Device Pixel vs. CSS pixel
- Using relative measure gives better results on various devices

How to use ems and percents

<code><body></code>	Browser's default text size is usually 16 pixels
<code><p></code>	100% or 1em is 16 pixels
<code><h3></code>	125% or 1.125em is 18 pixels
<code><h2></code>	150% or 1.5em is 24 pixels
<code><h1></code>	200% or 2em is 32 pixels

/ using 16px scale */*

```
body { font-size: 100%; }  
h3 { font-size: 1.125em; } /* 1.25 x 16 = 18 */  
h2 { font-size: 1.5em; }  /* 1.5 x 16 = 24 */  
h1 { font-size: 2em; }    /* 2 x 16 = 32 */
```

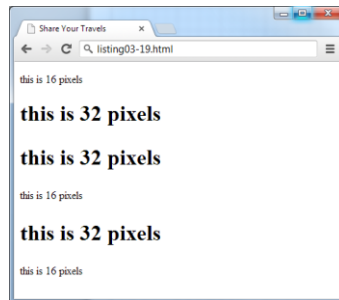
`<body>`

```
<p>this will be about 16 pixels</p>  
<h1>this will be about 32 pixels</h1>  
<h2>this will be about 24 pixels</h2>  
<h3>this will be about 18 pixels</h3>  
<p>this will be about 16 pixels</p>  
</body>
```

ems and percents complications

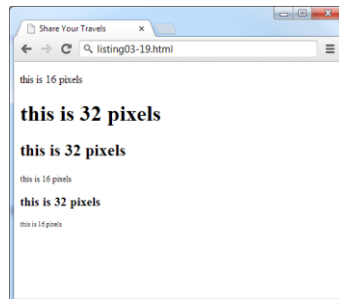
```
<body>
  <p>this is 16 pixels</p>
  <h1>this is 32 pixels</h1>
  <article>
    <h1>this is 32 pixels</h1>
    <p>this is 16 pixels</p>
    <div>
      <h1>this is 32 pixels</h1>
      <p>this is 16 pixels</p>
    </div>
  </article>
</body>
```

Percents and em units are relative to their parents, so if the parent font size changes, this affects all of its contents.



```
/* using 16px scale */
```

```
body { font-size: 100%; }
p    { font-size: 1em; }    /* 1 x 16 = 16px */
h1   { font-size: 2em; }    /* 2 x 16 = 32px */
```

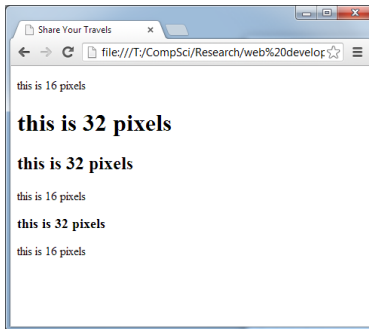


```
/* using 16px scale */
```

[illegible]

The rem unit

CSS3 supports a new relative measure, the **rem** (for root em unit). This unit is always relative to the size of the root element (i.e., the <html> element).



```
/* using 16px scale */
```

```
body { font-size: 100%; }  
p {  
    font-size: 16px; /* for older browsers: won't scale properly though */  
    font-size: 1rem; /* for new browsers: scales and simple too */  
}  
h1 { font-size: 2em; }  
  
article { font-size: 75% } /* h1 = 2 * 16 * 0.75 = 24px  
                           p = 1 * 16 = 16px */  
  
div { font-size: 75% } /* h1 = 2 * 16 * 0.75 * 0.75 = 18px  
                        p = 1 * 16 = 16px */
```

References

- W3C schools [<http://www.w3schools.com/>]
 - HTML tutorial
 - CSS tutorial
- Fundamentals of Web Developements
 - Chapter 2,3