



北京理工大学计算机学院  
School of Computer Science and Technology, BIT

# Web开发基础 HTML&CSS

赵刚

# 内容摘要

- **Basic HTML**
- Web Standards
- Basic CSS

# ● Hypertext Markup Language (HTML)

- 1993: Initial official proposed description of HTML submitted to the IETF standards group.
- 1995: HTML 2 becomes an official standard language by a publication called RFC 1866.
- 1996-97: HTML 3.2 standardizes various features including forms, tables, image maps, and internationalization.
- 1997: HTML 4 is proposed by W3C standard body, adding style sheets, scripting, frames, embedding objects, internationalization, and accessibility for disabilities.
- 1999: HTML 4.01 the last major version of the language is published by W3C. **A majority of pages on the Web today still use it as their started language.**
- 2001-01: XHTML, HTML based on XML
- HTML 5? Today?

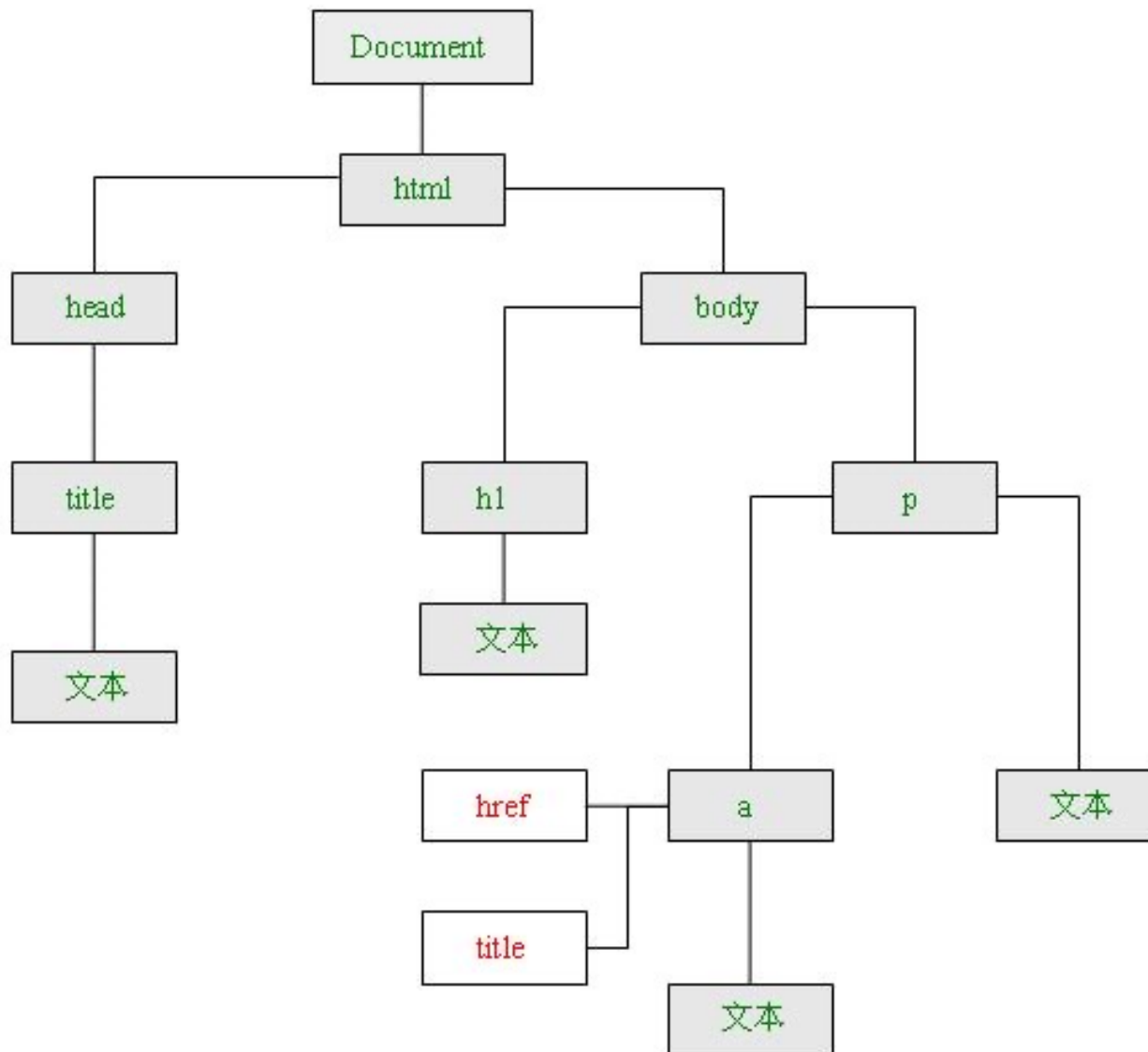


# ● Hypertext Markup Language (HTML)

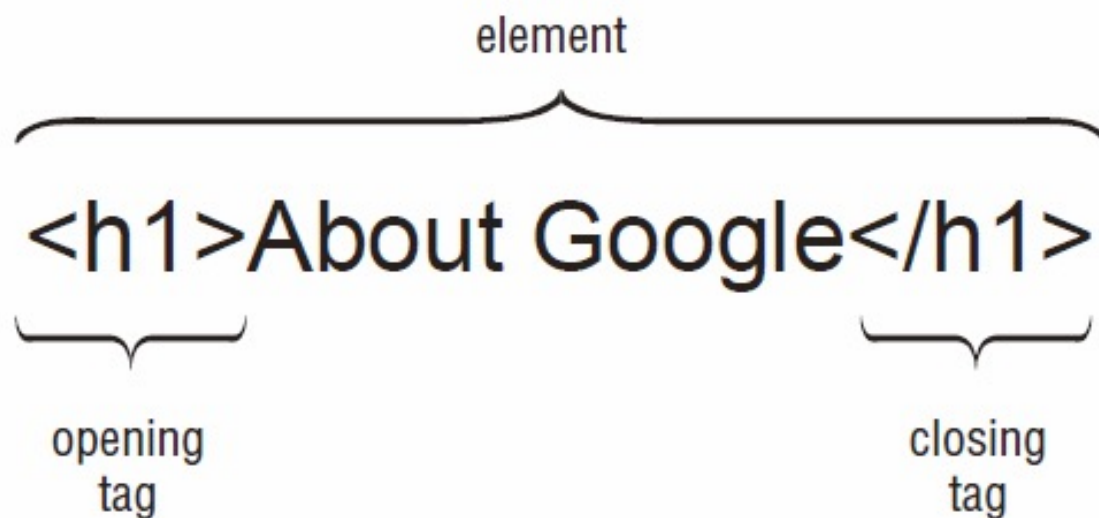
- describes the **content** and **structure** of information on a Web page
  - not the same as the presentation (appearance on screen)
- surrounds text contents with opening and closing **tags**
- each tag's name is called an **element**
  - syntax: `<element>content</element>`
  - example: `<p>This is an paragraph</p>`
- most whitespace is insignificant in HTML (ignored or collapsed to a single space)
- we will use a stricter, more standard version called [XHTML](#)

# ● Hypertext Markup Language (HTML)

- HTML coding convention: the structure of HTML is a tree.
  - indent nested elements
  - separate siblings with blank line when it makes reading easy.
- Responsibility of HTML languages
  - HTML describes the content and structure
  - Style Sheets (CSS) describes the appearance of the document
  - Script (JavaScript) describes the behavior of the document
- index.html



# *Tags and Elements*





# Comments: <!--...-->

- comments to document your HTML file or "comment out" text
- <!--My web page, by BIT Student CS 330, 2013 -->
- comments are still useful for disabling sections a page
- comments cannot be nested and cannot contain a -
- many web pages are not thoroughly commented (or at all)
  - comment is a communicative approach, to explain your designs and purposes to your colleagues, or even yourself sometime later.
  - comment is not for browsers of end users, but for developers and designers.



# The `<html>` Element

- The `<html>` element is the containing element for the whole XHTML document.

```
<html>
<head>
<title>Popular Websites: Google</title>
</head>
<body>
<h1>About Google</h1>
<p>Google is best known for its search engine, although
Google now offers a number of other services.</p>
<p>Google's mission is to organize the world's
information and make it universally accessible and
useful.</p>
<p>Its founders Larry Page and Sergey Brin started
Google at Stanford University.</p>
</body>
</html>
```

# Separating Heads from Bodies

- Whenever you write a web page in XHTML, the whole of the page is contained between the opening `<html>` and closing `</html>` tags, just as it was in the last example. Inside the `<html>` element, there are two main parts to the page:
- ☐ The `<head>` element: Often referred to as the head of the page, this contains information *about* the page
- ☐ The `<body>` element: Often referred to as the body of the page, this contains the information you *actually see* in the main browser window

# ***<head> Element***

- The <head> element is just a container for all other header elements
- Each <head> element should contain a <title> element indicating the title of the document, although it
- may also contain any combination of the other elements, in any order



# Page title: <title>

- describes the title of the Web page
- <title>Chapter 2: HTML Basics</title>
- placed within the head of the page
- displayed in the Web browser's title bar and when bookmarking the page

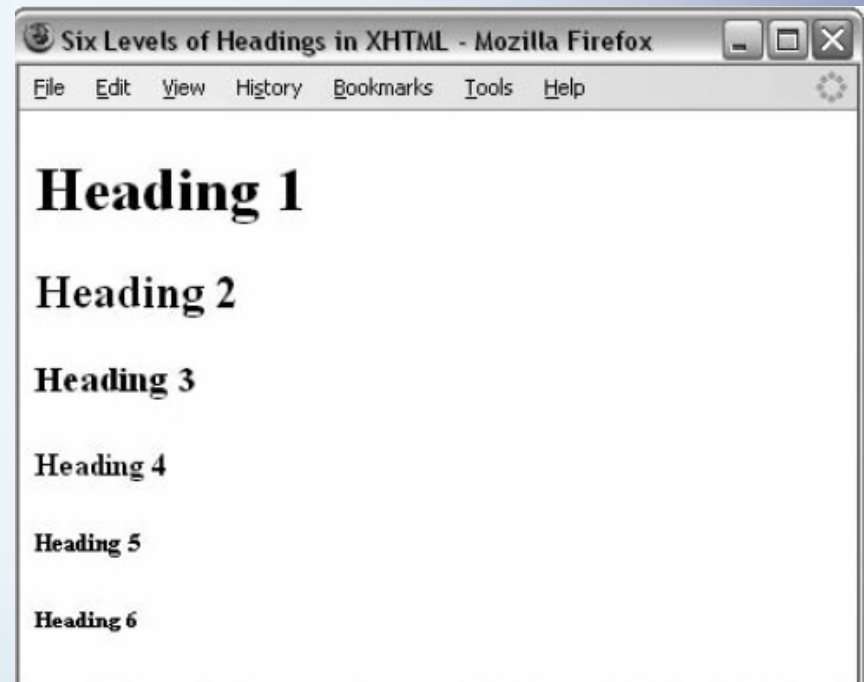
# ***<body> Element***

- The `<body>` element appears after the `<head>` element and contains the part of the web page that you actually see in the main browser window



# *Creating Headings Using hn Elements*

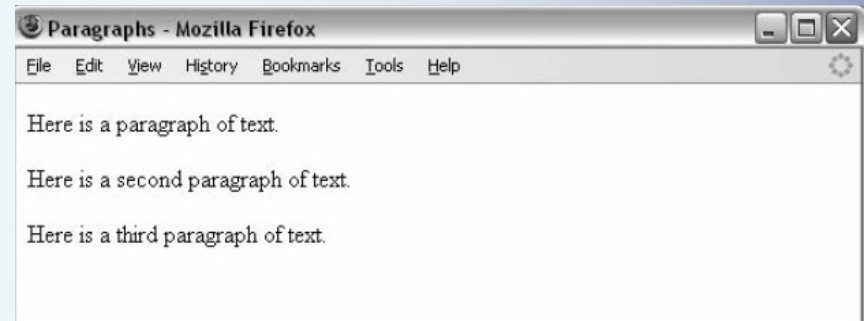
XHTML offers six levels of headings, which use the elements `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, and `<h6>`





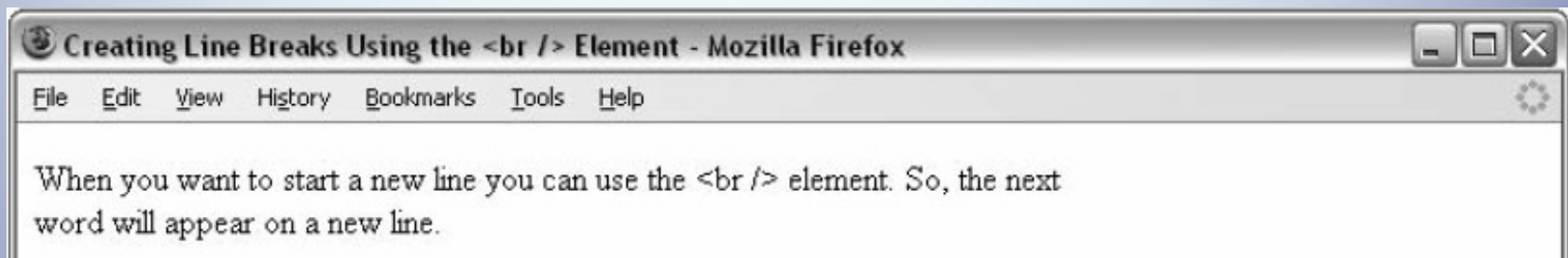
# Creating Paragraphs Using the `<p>` Element

- `<p>Here is a paragraph of text.</p>`
- `<p>Here is a second paragraph of text.</p>`
- `<p>Here is a third paragraph of text.</p>`



# ● *Creating Line Breaks Using the `<br />` Element*

- `<p>`When you want to start a new line you can use the `&lt;br /&gt;` element.
- So, the next`<br />`word will appear on a new line.`</p>`



# *Other* Presentational Elements

- `<b>` **bold**
- `<i>` *italicized text*
- `<u>` underlined
- `<s>` strikethrough
- `<hr />` creates a horizontal rule across the page
- `<em>` `<strong>` **Adds Strong Emphasis**
- ...



# Lists

- You can create three types of lists in XHTML:
  - ☐ **Unordered lists, which are like lists of bullet points**
  - ☐ **Ordered lists, which use a sequence of numbers or letters instead of bullet points**
  - ☐ **Definition lists, which allow you to specify a term and its definition**

# Using the `<ul>` Element to Create Unordered Lists

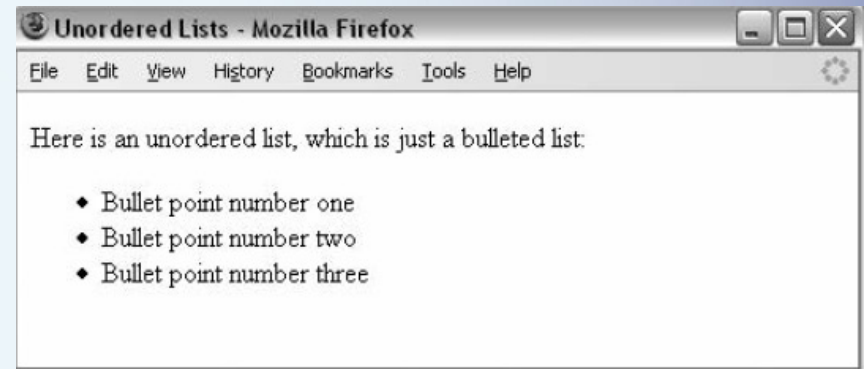
`<ul>`

`<li>Bullet point number one</li>`

`<li>Bullet point number two</li>`

`<li>Bullet point number three</li>`

`</ul>`



# Ordered Lists

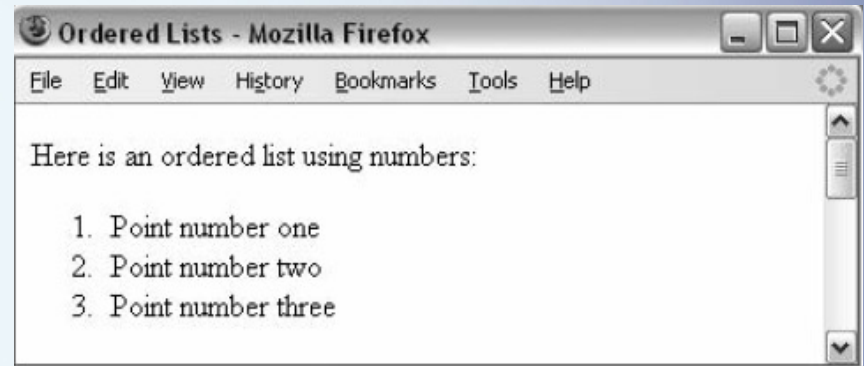
`<ol>`

`<li>Point number one</li>`

`<li>Point number two</li>`

`<li>Point number three</li>`

`</ol>`



## Type Attribute



# Definition Lists

`<dl>`

`<dt>Unordered List</dt>`

`<dd>A list of bullet points.</dd>`

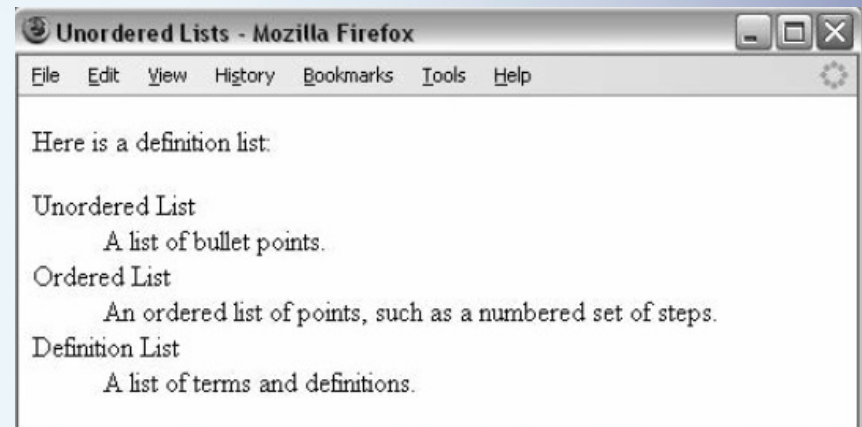
`<dt>Ordered List</dt>`

`<dd>An ordered list of points, such as a  
numbered set of steps.</dd>`

`<dt>Definition List</dt>`

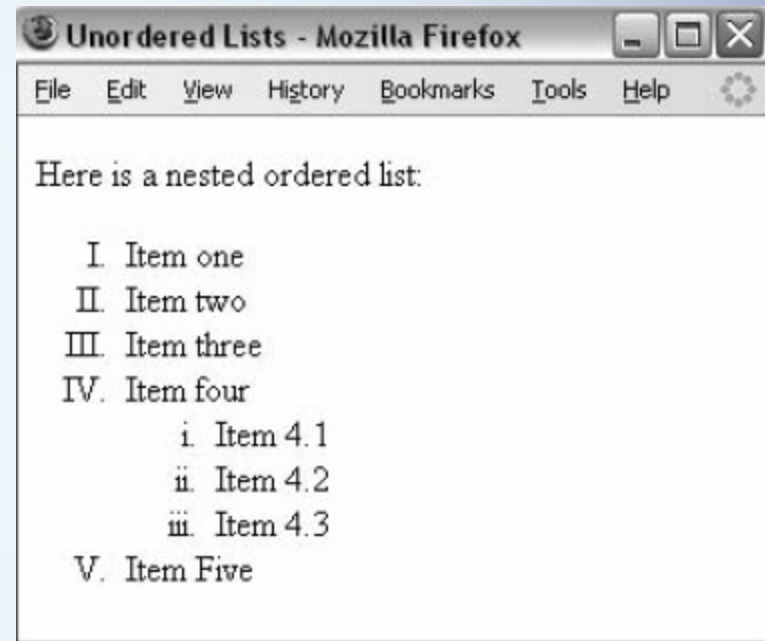
`<dd>A list of terms and definitions.</dd>`

`</dl>`



# Nesting Lists

```
<ol type="I">  
<li>Item one</li>  
<li>Item two</li>  
<li>Item three</li>  
  <li>Item four  
    <ol type="i">  
      <li>Item 4.1</li>  
      <li>Item 4.2</li>  
      <li>Item 4.3</li>  
    </ol>  
  </li>  
<li>Item Five</li>  
</ol>
```



# Links and Navigation

- A link is specified using the `<a>` element. Anything between the opening `<a>` tag and the closing `</a>` tag becomes part of the link a user can click in a browser.



# Linking to Other Documents

`<a href="http://www.Google.com/" title="Search the Web with Google">Google</a>`  
is a very popular search engine.



# *Linking to E-mail Addresses*

- `<a href="mailto:name@example.com">name@example.com</a>`

# ***Creating a Destination Anchor Using the name and id Attributes (linking to a specific part of a page)***

```
<h1>Linking and Navigation</h1>
```

```
<h2><a id="URL">URLs</a></h2>
```

```
<h2><a id="SourceAnchors">Source  
  Anchors</a></h2>
```

```
<h2><a  
  id="DestinationAnchors">Destination  
  Anchors</a></h2>
```

```
<h2><a  
  id="Examples">Examples</a></h2>
```

```
<p>This page covers the following topics:
```

```
<ul>
```

```
<li><a href="#URL">URLs</a></li>
```

```
<li><a href="#SourceAnchors">Source  
  Anchors</a></li>
```

```
<li><a  
  href="#DestinationAnchors">Destination  
  Anchors</a></li>
```

```
<li><a href="#Examples">Examples</a></li>
```

```
</ul>
```

```
</p>
```



# Images and Objects

- ☐ **GIF: Graphics Interchange Format**
- ☐ **JPEG: Joint Photographic Experts Group Format**
- ☐ **PNG: Portable Network Graphics**
- ☐ Flash movies are played with the Flash Player;
- ☐ Windows Media Files require Windows Media Player;
- ☐ MP3s can be played in various players including Flash Player, Windows Media Player, and QuickTime Player.

# ● *Adding Images Using the <img> Element*

- ``
- `src="url"`
- `alt="BIT logo"`



# *Adding a Flash Movie To A Page*

```
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" width="300"
height="200"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.c
ab">
<param name="movie" value="motion/flash_sample.swf">
<param name="play" value="true">
<param name="loop" value="false">
<embed src="motion/flash_sample.swf" width="300" height="200" play="true"
loop="false" QUALITY="best" menu="false" type="application/x-shockwave-flash"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod
_Version
=ShockwaveFlash">
</embed>
</object>
```



# Using Images as Links

```
<a href="../index.html" title="Click here to return to the  
home page">
```

```
</a>
```

# Tables

- create a table in XHTML using the `<table>` element
- A row is contained inside a `<tr>` element
- each cell is then written inside the row element using a `<td>` element
- If a row or column should contain a heading, a `<th>` element is used

<table border="1">

<tr>

<td>Row 1, Column 1</td>

<td>Row 1, Column 2</td>

</tr>

<tr>

<td>Row 2, Column 1</td>

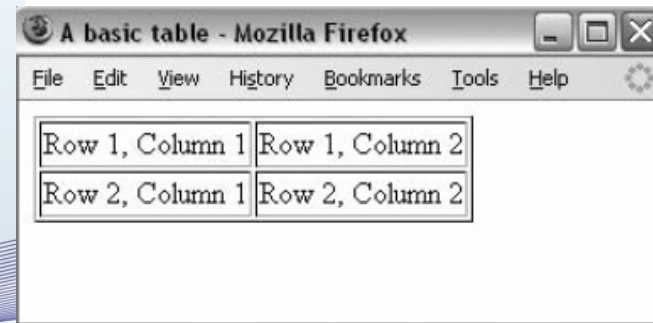
<td>Row 2, Column 2</td>

</tr>

</table>

A diagram of a 5x4 table. The table is labeled 'ROWS' at the top and 'COLUMNS' on the left. The table contains the following data:

Column 1 Row 1	Column 2 Row 1	Column 3 Row 1	Column 4 Row 1
Column 1 Row 2	Column 2 Row 2	Column 3 Row 2	Column 4 Row 2
Column 1 Row 3	Column 2 Row 3	Column 3 Row 3	Column 4 Row 3
Column 1 Row 4	Column 2 Row 4	Column 3 Row 4	Column 4 Row 4
Column 1 Row 5	Column 2 Row 5	Column 3 Row 5	Column 4 Row 5







# Grouping Elements with `<div>` and `<span>`

- The `<div>` and `<span>` elements allow you to group several elements to create sections or subsections of a page
- `<div>` multi-line
- `<span>` within-line

# <div> <span>

```
<div class="footnotes">  
<h2>Footnotes</h2>  
<p><b>1</b> The World Wide Web was  
  invented by Tim Berners-Lee</p>  
<p><b>2</b> The W3C is the World  
  Wide Web Consortium who maintain  
  many Web standards</p>  
</div>
```

```
<div class="footnotes">  
<h2>Footnotes</h2>  
<p><span class="inventor"><b>1</b> The  
  World Wide Web was invented by Tim  
  Berners Lee</span></p>  
<p><b>2</b> The W3C is the World Wide Web  
  Consortium who maintain many Web  
  standards</p>  
</div>
```



# Forms

- With javascript&ASP

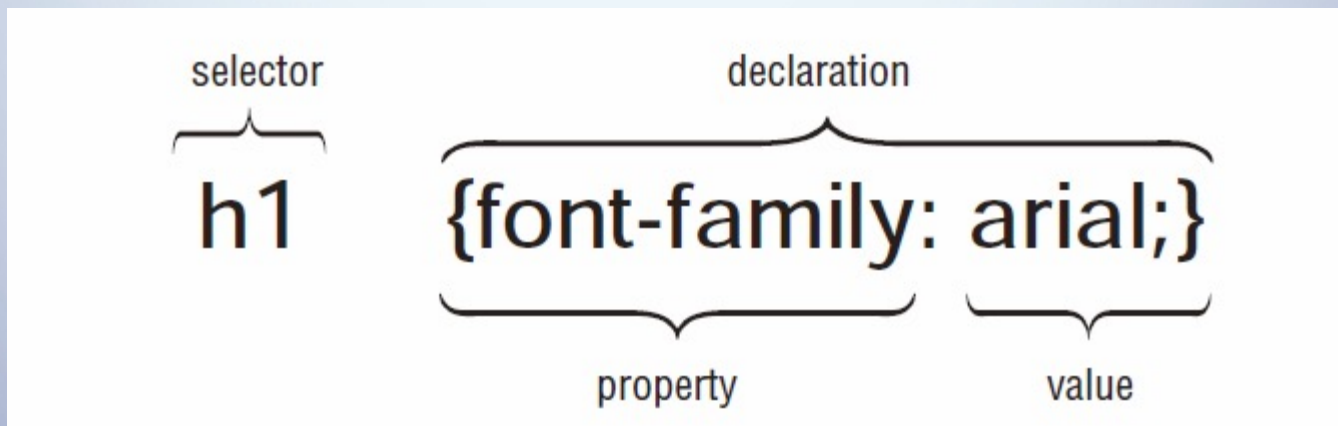


# Cascading Style Sheets(CSS)

- CSS should be used to control the appearance of web pages
  - ☐ What makes up a CSS rule
  - ☐ How to place CSS rules within your document, and how to link to an external CSS document
  - ☐ How properties and values control presentation of different elements within your document
  - ☐ How to control the presentation of text using CSS

# Introducing CSS

- ❑ The *selector*, which indicates which element or elements the declaration applies to (if it applies to more than one element, you can have a comma-separated list of several elements)
- ❑ The *declaration*, which sets out how the elements should be styled



# example of a CSS

```
h1, h2, h3 {  
    font-weight:bold;  
    font-family:arial, verdana, sans-serif;  
    color:#000000;  
    background-color:#FFFFFF;}
```



# *Inheritance*

- One of the powerful features of CSS is that many of the properties that have been applied to one element will be *inherited by child elements (elements contained within the element that the rules were declared upon)*

# Where You Can Add CSS Rules

- ❑ used a separate style sheet, or *external style sheet*, to contain the CSS rules
- ❑ Inside the <head> element, contained with a <style> element
- ❑ As a value of a style attribute on any element that can carry the style attribute

```
<head>
<title>CSS Example</title>
<link rel="stylesheet" type="text/css" href="ex01.css" />
</head>
```

```
<head>
<title>Internal Style sheet</title>
<style type="text/css">
body {
color:#000000;
background-color:#ffffff;
font-family:arial, verdana, sans-serif; }
h1 {font-size:18pt;}
p {font-size:12pt;}
</style>
</head>
```

```
<td style="font-family:courier; padding:5px; border-
style:solid;border-width:1px; border-color:#000000;">
```



# ***Advantages of External CSS Style Sheets***

- The same style sheet can be *reused by all of the web pages in your site.*
- *No need redownload from server*
- You can change the appearance of several pages by altering just the style sheet rather than each individual page
- The style sheet can act as a style template to help different authors
- different style sheets can be attached to the same document provided to different device
- Code reuse



# CSS Properties

- [www.devguru.com/Technologies/css/quickref/css\\_index.html](http://www.devguru.com/Technologies/css/quickref/css_index.html)
- [www.w3schools.com/css/css\\_reference.asp](http://www.w3schools.com/css/css_reference.asp)

# Specifying colors

- `p { color: red;}`
- `h2 { color: rgb(128, 0, 196);}`
- `h4 { color: #FF8800;}`
- RGB codes: red, green, and blue values from 0 (none) to 255 (full)
- hex codes: RGB values in base-16 from 00 (0, none) to FF (255, full)



# CSS properties for fonts

<u>Property</u>	<u>description</u>
font-family	which font will be used
font-size	how large the letters will be drawn
font-style	used to enable/disable italic style
font-weight	used to enable/disable bold style



# font-family

- `p { font-family: Georgia; }`
- `h2 { font-family: "Courier New"; }`
- Font of Chinese Characters
  - most of browsers only support SimSun ("宋体")
  - IE can support other fonts Windows OS supported
  - 黑体:SimHei 新宋体:NSimSun 仿宋体:FangSong SimFang?
  - 楷体:KaiTi SimKai? 仿宋\_GB2312:FangSong\_GB2312
  - 楷体\_GB2312:KaiTi\_GB2312 微软雅黑体:Microsoft YaHei
  - ...

# More about font-family

- `p { font-family: Garamond, "Times New Roman", serif; }`
  - If no Garamond then uses TNR, and then uses serif. enclose multi-word font names in quotes
  - can specify multiple fonts from highest to lowest priority
- generic font names:
  - serif, sans-serif, monospace, cursive, marlett

# font-size

- `p { font-size: 20pt; }`
- units: pixels (px) vs. point(pt) vs. m-size(em)
  - 16px, 16pt, 1.16em(\*related to default text size)
- vague font size:
  - xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger
- percentage font sizes
  - e.g.: 90%, 120%



# font-weight,font-style

- `p { font-weight: bold;  
font-style: italic;}`
- either of the above can be set to normal to turn them off

# Grouping styles

- `p,h1,h2{ color:green;}`  
`h2{ background-color:yellow;}`

*a style can select multiple elements separated by commas*

- the individual elements can also have their own style (like h2 above)

# CSS comments: `/*...*/`

- `/* This is a coment. It can span many lines in the CSS file. */`  
`p{ color:red; background-color: aqua;}`
- the `//` single-line comment style is NOT supported in CSS
- the `<!--...-->` HTML comment style is also NOT supported in CSS



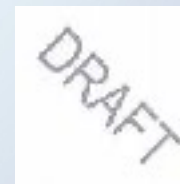
# Body Styles

- `body{ font-size: 16px; }`
- to apply a style to the entire body of your page, write a selector for the body element
- saves you from manually applying a style to each element

# CSS properties for backgrounds

<u>property</u>	<u>description</u>
background-color	color to fill background
background-image	image to place in background
background-position	placement of bg image within element
background-repeat	whether/how bg image should be repeated
background-attachment	whether bgimage scrolls with page

```
body {  
    background-image: url("images/draft.jpg");  
}
```



# Selectors

- ***Universal Selector*** \*{}
- ***The Type Selector*** h1, h2, p {}
- ***The Class Selector***


<p class="BackgroundNote">This paragraph contains an aside.</p>

.BackgroundNote {}

p.BackgroundNote {}

- ***The ID Selector***
  - p#abstract





# *pseudo-classes*

## **Pseudo-class   Purpose**

link	Styles for links in general
visited	Styles for links that have already been visited
active	Styles for links that are currently active (being clicked)
hover	Styles for when someone is hovering over a link

```
body {background-color:#ffffff;}
a {
font-family: arial, verdana, sans-serif;
font-size:12px;
font-weight:bold;}
a:link {
color:#0000ff;
text-decoration:none;}
a:visited {
color:#333399;
text-decoration:none;}
a:active {
color:#0033ff;
text-decoration:underline;}
a:link:hover {
background-color:#e9e9e9;
text-decoration:underline;}
```

# Lab 2 Basic HTML and CSS

- **Purpose**

- The purpose of this lab is to practice writing basic web pages with HTML and CSS

- **Exercise 1: About Me Page**

- Create a page named aboutme.html that describes you.

- Your name
    - A description of yourself in two sentences or less. Emphasize the most important word(s) by putting them in bold.
    - A list of classes you are taking right now at BIT.
    - Your 3 favorite movies, books, or TV shows, in order. Make at least one link to an interesting site about that tv show/movie/book, such as its [IMDB](#) page.
    - Two images, one that represents you when you're happy and the other to represent you when you're sad. (These can be any images you like. Consider searching for images on [Google Image Search](#).)
    - Something interesting about one or more of your neighbors (people sitting at computers next to you)



## • Exercise 2: Style Your Page with CSS

- Create a stylesheet named aboutme-style.css to improve the appearance of your About Me page. Your stylesheet should do the following without any modification to your HTML code:
  - Change the color of at least two elements
  - Change the font properties of at least two elements (such as family, size, weight, style). Here are some standard fonts you may want to use:  
Arial, Arial Black, Verdana, Trebuchet MS, Georgia, Tahoma, Courier New, Times New Roman
  - Change at least one other thing of your choosing (such as background color, text alignment, etc.)



## About Xiaocheng

My name is Xiaocheng Chen and I am a **optimistic**, **sunshine**, and **hunsand** boy.

### My Favorite Movies

*I have watched lots of movies and my favorite are as followed.*

1. Forrest Gump([IMDB](#))
2. Saving Private Rayn([IMDB](#))
3. The sound of music([IMDB](#))

### My Habits

- Play Basketball
- Play Badminton/Shuttlecock
- Swimming

### My Moods



Happy:



Sad:

### My Best Friends

- Tauruser: A smart boy good at programming.
- Eric: Omnipotence is a word that describe him.

## • Exercise 3: Validate Your Page and CSS

validate your HTML and CSS code to make sure they match the strict XHTML 1.1 specifications. To run the validators, do the following:

1. Open <http://validator.w3.org/> (for XHTML) or <http://jigsaw.w3.org/css-validator/> (for CSS)
2. Click Browse in the Validate by File Upload section
3. Locate the local file and click Open
4. Click Check

If there are any errors, fix the first one, then repeat steps 1 through 4.





## • Exercise 4 (advanced): Advanced Style Techniques

- If you manage to complete the exercises before lab time is up, edit your stylesheet to also do the following:
  - Your hyperlinks should not have an underline by default. When a user hovers over the link, the underline should appear.  
*Hint:* The property you are looking for is a type of CSS pseudo-class.