

CPIS 358: INTERNET APPLICATIONS & WEB PROGRAMMING

CREATE

BEAUTIFUL

WEBSITES

GO>

CHAPTER 3-PART 1: INTRODUCTION TO CASCADING STYLE SHEETS™ (CSS)

TEXTBOOK

2

- You can read Chapter 4 page 136-->173

AGENDA

3

- Introduction
- Inline Styles
- Embedded Style Sheets
- Linking External Style Sheets
- Positioning Elements
- Backgrounds
- Element Dimensions
- Box Model and Text Flow
- Media Types
- Building a CSS Drop-Down Menu
- Conflicting Styles

OBJECTIVES

4

- At the end of this chapter, the student should be able to:
 - ▣ control the appearance of a website by creating style sheets
 - ▣ use a style sheet to give all the pages of a website the same look and feel
 - ▣ use the class attribute to apply styles
 - ▣ specify the precise font, size, color and other properties of displayed text
 - ▣ specify element backgrounds and colors
 - ▣ understand the box model and how to control margins, borders and padding
 - ▣ use style sheets to separate presentation from content

Introduction

5

- Cascading Style Sheets (CSS)
 - ▣ Used to specify the presentation of elements separately from the structure of the document
- CSS validator
 - `jigsaw.w3.org/css-validator/`
- Different ways to define styles
 - ▣ Inline styles
 - ▣ Embedded style sheets
 - ▣ External style sheets

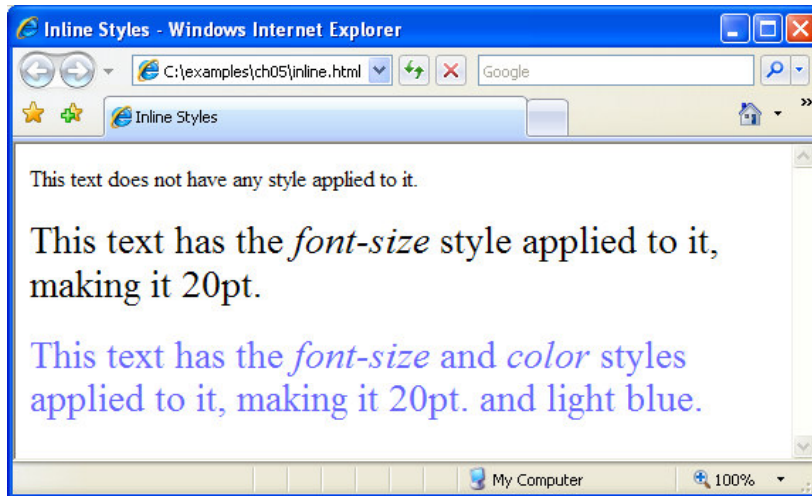
Inline Styles...

6

- Inline style
 - ▣ declares a style for an individual element by using the `style` attribute in the element's start tag
 - ▣ Syntax
 - `<HTML_Element style="property1:value1; property2:value2;...">`
- Each CSS property is followed by a colon and the value of the attribute
 - ▣ Multiple property declarations are separated by a semicolon
 - ▣ For example: `color` property sets text color
 - Color names and hexadecimal codes may be used as the value

Inline Styles...

7



```
<!DOCTYPE html>

<!-- Fig. 4.1: inline.html -->
<!-- Using inline styles -->
<html>
  <head>
    <meta charset = "utf-8">
    <title>Inline Styles</title>
  </head>
  <body>
    <p>This text does not have any style applied to it.</p>

    <!-- The style attribute allows you to declare -->
    <!-- inline styles. You should separate multiple -->
    <!-- style properties with a semicolon. -->
    <p style = "font-size: 20pt;">This text has the
      <em>font-size</em> style applied to it, making it 20pt.
    </p>

    <p style = "font-size: 20pt; color: deepskyblue;">
      This text has the <em>font-size</em> and
      <em>color</em> styles applied to it, making it
      20pt and deep sky blue.</p>
  </body>
</html>
```

Style attribute

Sets the paragraph's
font size to 20pt

Sets the paragraph's
color to light blue

Good Programming Practice 1

Inline styles do not truly separate presentation from content. To apply similar styles to multiple elements, use embedded style sheets or external style sheets, introduced later in this chapter

Embedded and External Style Sheets...

9

- The purpose of those styles is to reuse the same style to different elements
- Embedded style Sheets are defined in the head section of HTML's document
- External Style Sheets are defined in a separate document . CSS
- Both of them are a set of rules that describe how HTML's Elements will appear

Embedded style sheets

10

- Defined in the head of the HTML document
- Syntax
- The head specifies the type of style sheet. The style element's type attribute specifies the MIME type (Multipurpose Internet Mail Extensions) of the style sheet
- The body

```
1  <!DOCTYPE html>
2
3  <html>
4    <head>
5      <style type = MIME>
6        rule1
7        rule2
8
9
10
11      rulen
12    </style>
13  </head>
```

Embedded Style Sheets...

11

- The Style Sheet's body is a set of rules
 - ▣ Each rule consists of a selector followed by a declaration block
 - A selector can be: HTML's Element, Class, Pseudo Class
 - A declaration block is a set of properties and their values {property:value...}
 - ▣ Syntax
 - Selector {property:value...}

Embedded Style Sheets...

12

□ Selector

▣ HTML's Element: body, em, p, h1 ...

- The style defined will be applied only to this element

▣ Class

- The **style-class** declarations
 - are preceded by a period (.)
 - Can be applied to any elements

▣ Pseudo-class

- Is used in order to change the link's color according to its state (not visited, visited, hover, active; have a focus),
- The declaration order should be **:link**, **:visited**, **:hover**, **:active** and **:focus**
 - **:hover** pseudoclass
 - used to apply styles to an element when the mouse cursor is over it
- Just to memorize this order: **LoVe HAte**

Embedded and External Style Sheet

13

- A declaration Block is a set of properties and its values {property:value...}

Property	Description	Values
font-weight	specifies the “boldness” of text	Bold, normal, bolder (bolder than bold text), lighter
font-family	Generic font families allow authors to specify a type of font instead of a specific font, in case a browser does not support a specific font	Tahoma...
font-size	specifies the size used to render the font	12, 14 pt...

Embedded Style Sheets...

14

- This code creates embedded style sheet containing four styles

```
<html>
<head>
  <meta charset = "utf-8">
  <title>Embedded Style Sheet</title>

  <!-- this begins the style sheet section -->
  <style type = "text/css">
    em      { font-weight: bold;
              color: black; }
    h1      { font-family: tahoma, helvetica, sans-serif; }
    p       { font-size: 12pt;
              font-family: arial, sans-serif; }
    .special { color: purple; }
  </style>
</head>
...
```

Embedded Style Sheets...

15

Style sheet begins

Sets the MIME type to text/css

Sets the properties for all elements in the document within em tags

Sets the properties for all h1 elements in the document

Sets the properties for all p elements in the document

Style sheet ends

Creates a special class

The special class is applied to this p element

```
1 <!DOCTYPE html>
2
3 <!-- Fig. 4.3: embedded.html -->
4 <!-- Embedded style sheets. -->
5 <html>
6   <head>
7     <meta charset = "utf-8">
8     <title>Embedded Style Sheet</title>
9
10    <!-- this begins the style sheet section -->
11    <style type = "text/css">
12      em { font-weight: bold;
13          color: black; }
14      h1 { font-family: tahoma, helvetica, sans-serif; }
15      p { font-size: 12pt;
16         font-family: arial, sans-serif; }
17      .special { color: purple; }
18    </style>
19  </head>
20  <body>
21    <!-- this attribute applies the .special style class -->
22    <h1 class = "special">Deitel & Associates, Inc.</h1>
23
24    <p>Deitel & Associates, Inc. is an authoring and corporate training organization specializing in
25      programming languages, Internet and web technology, iPhone and Android app development, and object
26      technology education.</p>
27
28    <h1>Clients</h1>
29    <p class = "special">The company's clients include many
30      <em>Fortune 1000 companies</em>, government agencies,
31      branches of the military and business organizations.</p>
32  </body>
33 </html>
```

Embedded Style Sheets

16

Deitel & Associates, Inc.

Deitel & Associates, Inc. is an authoring and corporate training organization specializing in programming languages, Internet and web technology, iPhone and Android app development, and object technology education.

Clients

The company's clients include many *Fortune 1000 companies*, government agencies, branches of the military and business organizations.

Linking External Style Sheets...

17

- External style sheets are separate documents that contain only CSS rules
- Help create a uniform look for a website
 - ▣ separate pages can all use the same styles
 - ▣ When changes to the styles are required, you need to modify only a single CSS file to make style changes across all the pages that use those styles
 - This concept is known as skinning

Always use an external style sheet when developing a website with multiple pages. External style sheets separate content from presentation, allowing for more consistent look-and-feel, more efficient development, and better performance.

Linking External Style Sheets...

19

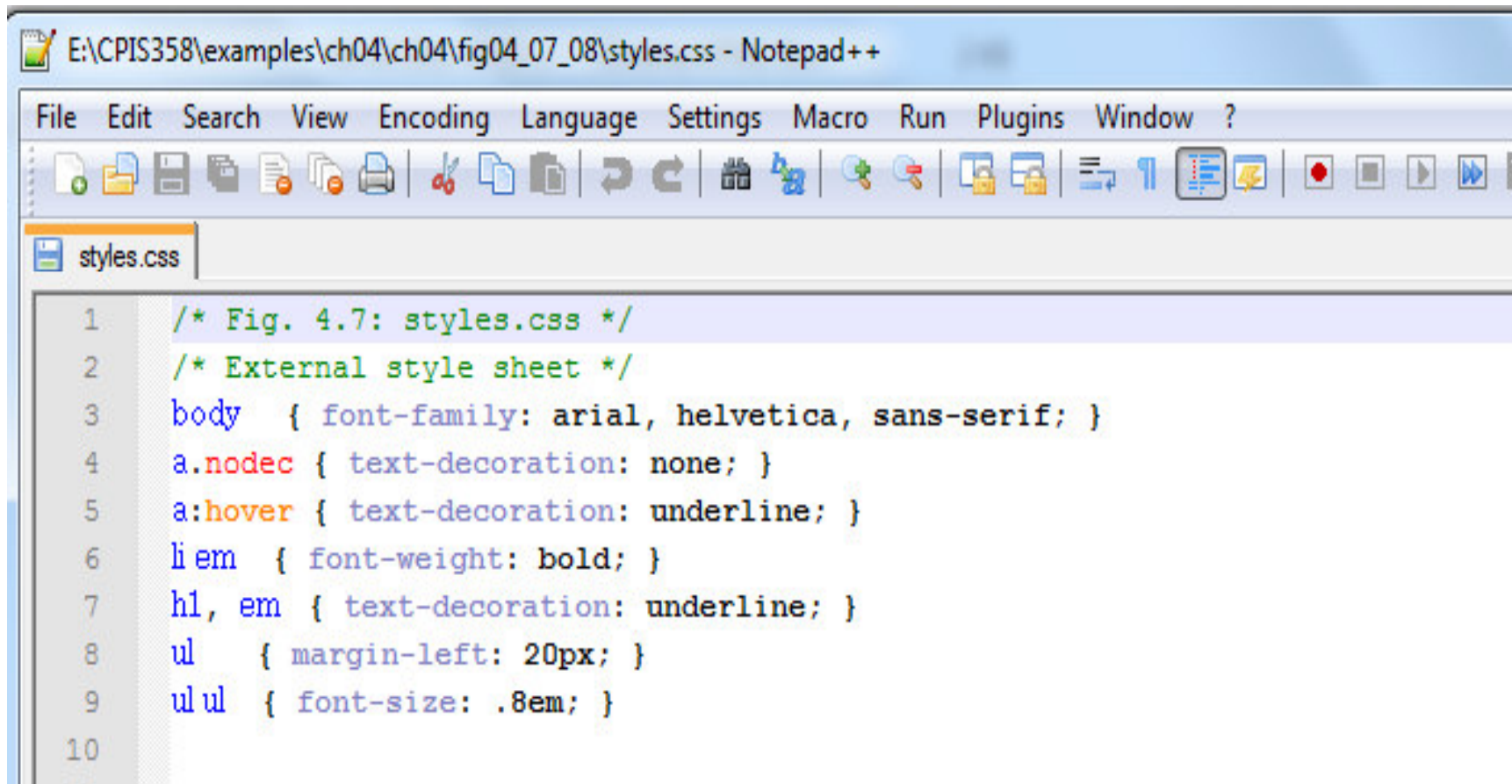
- How to use External style sheets?

a.css	b.html
Rule1 Rule2 ru1en	<head> <link rel="stylesheet" Type="text/css" Href="a.css"> </head>

- ▣ You should define a file.css that contains all styles
- ▣ In your HTML script, you refer to CSS's file by using
 - link element
 - Uses rel attribute to specify a relationship between two documents
 - rel attribute declares the linked document to be a stylesheet for the document
 - type attribute specifies the MIME type of the related document
 - href attribute provides the URL for the document containing the style sheet

Example of External style sheet: CSS file

20



```
1  /* Fig. 4.7: styles.css */
2  /* External style sheet */
3  body { font-family: arial, helvetica, sans-serif; }
4  a.nodect { text-decoration: none; }
5  a:hover { text-decoration: underline; }
6  li em { font-weight: bold; }
7  h1, em { text-decoration: underline; }
8  ul { margin-left: 20px; }
9  ul ul { font-size: .8em; }
10
```

Example of External style sheet: HTML document...

21

```
1 <!DOCTYPE html>
2
3 <html>
4   <head>
5     <meta charset="utf-8">
6     <title>Linking External Style Sheets</title>
7     <link rel = "stylesheet" type = "text/css"
8       href = "styles.css">
9   </head>
10  <body>
11    <h1>Shopping list for <em>Monday</em>:</h1>
12
13    <ul>
14      <li>Milk</li>
15      <li>Bread
16        <ul>
17          <li>White bread</li>
18          <li>Rye bread</li>
19          <li>Whole wheat bread</li>
20        </ul>
21      </li>
22      <li>Carrots</li>
23      <li>Yogurt</li>
24      <li>Pizza <em>with mushrooms</em></li>
25    </ul>
26
27    <p><em>Go to the</em>
28      <a class = "nodec" href = "http://www.deitel.com">
29        Grocery store</a>
30    </p>
31  </body>
32 </html>
33
```

The linked document is declared to be the current one's stylesheet

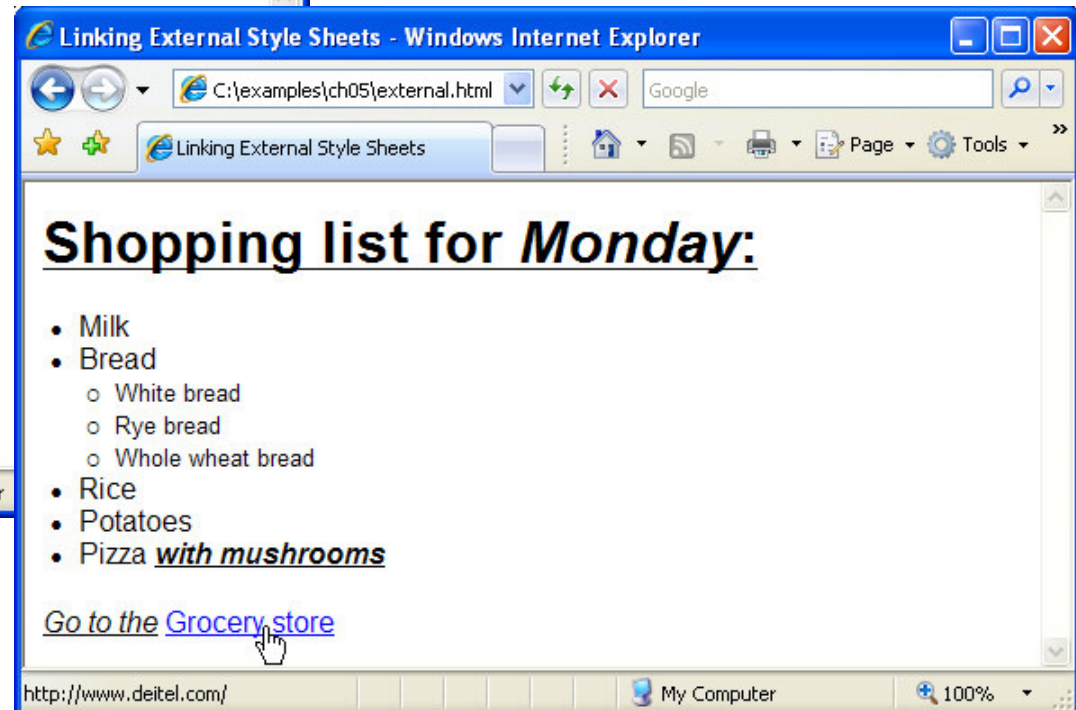
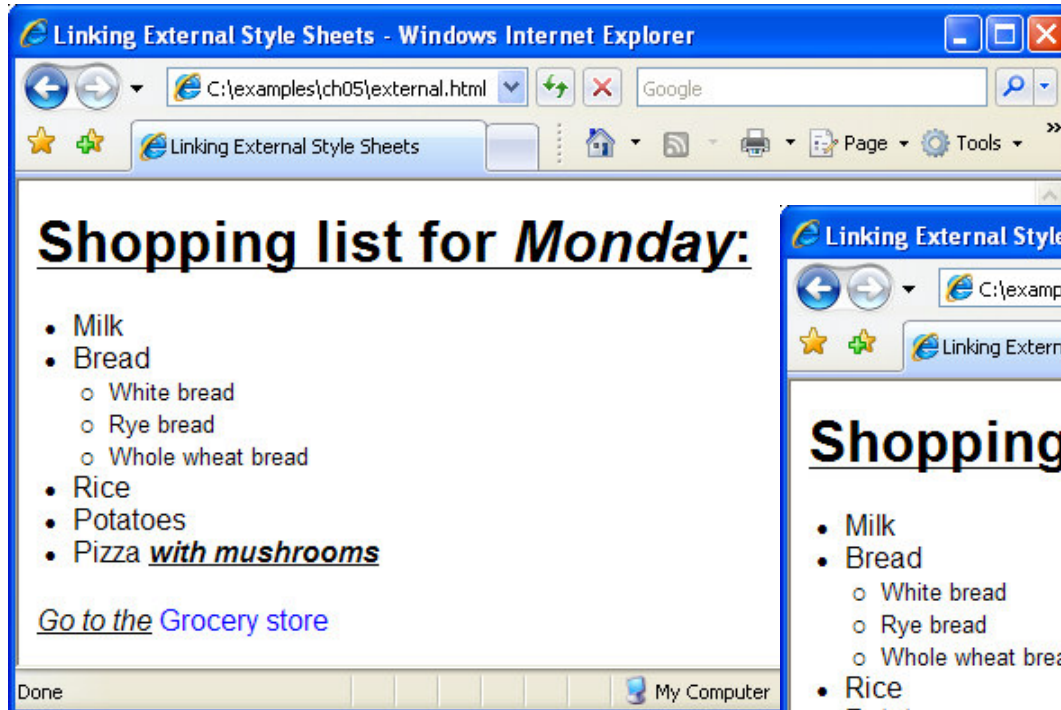
The linked document's MIME type is text/css

The linked document's URL is styles.css

Hyper Text Markup Language file length : 1854 lines : 49 Ln : 8 Col : 30 Sel : 0 Dos\Windows ANSI INS

Example of External style sheet: HTML document

22



External style sheets are reusable. Creating them once and reusing them

- reduce programming effort
- reduce load time and bandwidth usage on a server, since the style sheet can be downloaded once, stored by the web browser, and applied to all pages on a website.

Rules to respect when we define Embedded or External style sheets ...

- To apply rules to multiple elements, separate the elements with commas in the style sheet
 - `li, em{font-weight:bold}`
- To apply rules to only a certain type of element that is a child of another type, separate the element names with spaces
 - `li em {text-decoration: underline}`
- Pseudoclasses are separated by a colon (with no surrounding spaces) from the name of the element to which they are applied

Rules to respect when we define Embedded or External style sheets

- Relative length measurements:
 - ▣ px (pixels – size varies depending on screen resolution)
 - ▣ em (usually the height of a font's uppercase M)
 - ▣ ex (usually the height of a font's lowercase x)
 - ▣ Percentages (of the font's default size)
- Absolute-length measurements (units that do not vary in size):
 - ▣ in (inches)
 - ▣ cm (centimeters)
 - ▣ mm (millimeters)
 - ▣ pt (points; $1 \text{ pt} = 1/72 \text{ in}$)
 - ▣ pc (picas; $1 \text{ pc} = 12 \text{ pt}$)

Positioning Elements

26

- Normally, elements are positioned on the page in the order that they appear in the HTML document
- CSS `position` property offers the possibility to change the position of elements
- CSS `position` property allows
 - ▣ Absolute positioning
 - ▣ Relative positioning
 - ▣ Fixed positioning

Positioning Elements: Absolute Positioning, z-index...

27

- The absolute positioning provides greater control over where on a page elements reside
- Specifying an element's position as absolute removes it from the normal flow of elements on the page and positions it according to distance from the top, left, right or bottom margin of its parent element
- The z-index property allows a developer to layer overlapping elements
- Elements that have higher z-index values are displayed in front of elements with lower z-index values

Absolute positioning of elements...

28

The image shows a code editor with three tabs: `styles.css`, `external.html`, and `positioning.html`. The `positioning.html` tab is active, displaying the following HTML code:

```
1 <!DOCTYPE html>
2
3 <!-- Fig 4.9: positioning.html -->
4 <!-- Absolute positioning of elements. -->
5 <html>
6   <head>
7     <meta charset = "utf-8">
8     <title>Absolute Positioning</title>
9     <style type = "text/css">
10      .background_image { position: absolute;
11                          top: 0px;
12                          left: 0px;
13                          z-index: 1; }
14      .foreground_image { position: absolute;
15                          top: 25px;
16                          left: 100px;
17                          z-index: 2; }
18      .text { position: absolute;
19              top: 25px;
20              left: 100px;
21              z-index: 3;
22              font-size: 20pt;
23              font-family: tahoma, geneva, sans;
24      </style>
25   </head>
```

Five callout boxes provide explanations for specific CSS properties:

- Class that sets an element's absolute position at the top left of the containing element**: Points to the `position: absolute;` property in the `.background_image` rule.
- Lowest z-index, so this element is behind all the others**: Points to the `z-index: 1;` property in the `.background_image` rule.
- Set element's position 25px from the top and 100 from the left**: Points to the `top: 25px;` and `left: 100px;` properties in the `.foreground_image` rule.
- This element will appear on top of the first one, since it has a higher z-index**: Points to the `z-index: 2;` property in the `.foreground_image` rule.
- This element will appear on top of all others, since it has the highest z-index**: Points to the `z-index: 3;` property in the `.text` rule.

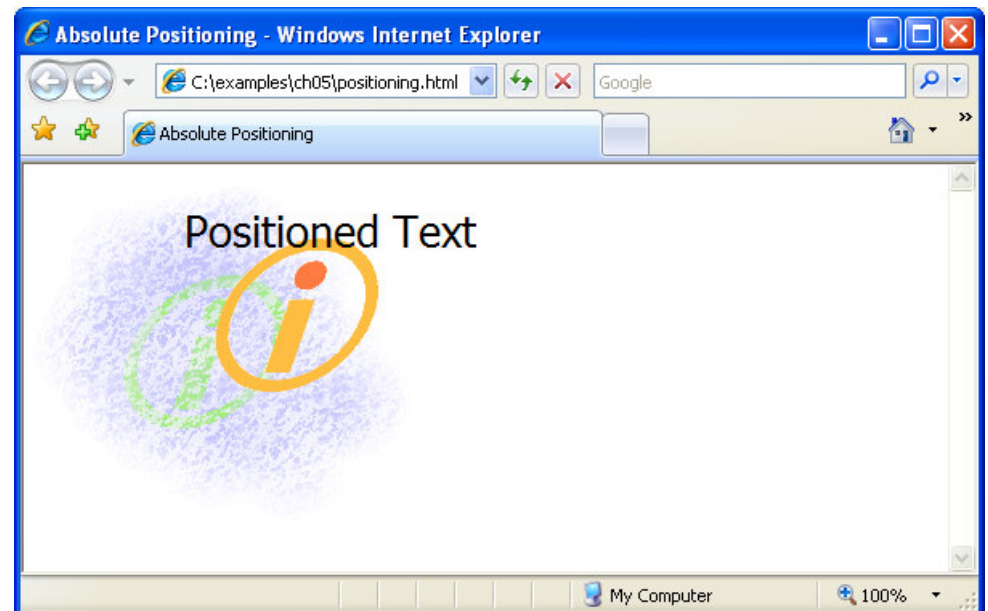
Absolute positioning of elements

29

```
<body>
  <p><img src = "background_image.png" class = "background_image"
    alt = "First positioned image" /></p>

  <p><img src = "foreground_image.png" class = "foreground_image"
    alt = "Second positioned image" /></p>

  <p class = "text">Positioned Text</p>
</body>
</html>
```



Positioning Elements: Relative

Positioning, span...

30

- Relative positioning keeps elements in the general flow on the page and offsets them by the specified top, left, right or bottom value
- So the positioning is relative to other elements in the flow

Positioning Elements : Relative

Positioning, span...

31

□ Inline-level elements

- ▣ Do not change the flow of the document

- ▣ Examples:

 - `img`

 - `a`

 - `em`

 - `strong`

 - `span`

 - Grouping element

 - Does not apply any formatting to its contents

 - Creates a container for CSS rules or `id` attributes to be applied to a section

Positioning Elements : Relative

Positioning, span...

32

- Block-level elements
 - ▣ Displayed on their own line
 - ▣ Have virtual boxes around them
 - ▣ Examples:
 - p
 - all headings (h1 through h6)
 - div
 - A grouping element like span

Span Vs. Div

33

Span

□ Code

```
<p><span>0000000<span>111111</span>222222</span></p>
```

□ Displaying

0000000111111222222

Div

□ Code

```
<div>0000000<div>111111</div>222222</div>
```

□ Displaying

0000000
111111
222222

Positioning Elements: Relative

Positioning, span...

```
34 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset = "utf-8">
5   <title>Relative Positioning</title>
6   <style type = "text/css">
7     p      { font-size: 1.3em;
8             font-family: verdana, arial, sans-serif; }
9     span   { color: red;
10            font-size: .6em;
11            height: 1em; }
12     .super { position: relative;
13             top: -1ex; }
14     .sub   { position: relative;
15             bottom: -1ex; }
16     .shiftleft { position: relative;
17                 left: -1ex; }
18     .shiftright { position: relative;
19                  right: -1ex; }
20   </style>
21 </head>
22 <body>
23   <p>The text at the end of this sentence
24     <span class = "super">is in superscript</span></p>
25
26   <p>The text at the end of this sentence
27     <span class = "sub">is in subscript</span></p>
28
29   <p>The text at the end of this sentence
30     <span class = "shiftleft">is shifted left</span></p>
```

Positions element 1 ex upwards

Positions element 1 ex downwards

Positions element 1 ex to the left

Positions element 1 ex to the right

Apply the super class to this span element

Apply the sub class to this span element

Positioning Elements: Relative

Positioning, span...

35

```
<body>
  <p>The text at the end of this sentence
    <span class = "super">is in superscript</span>.</p>

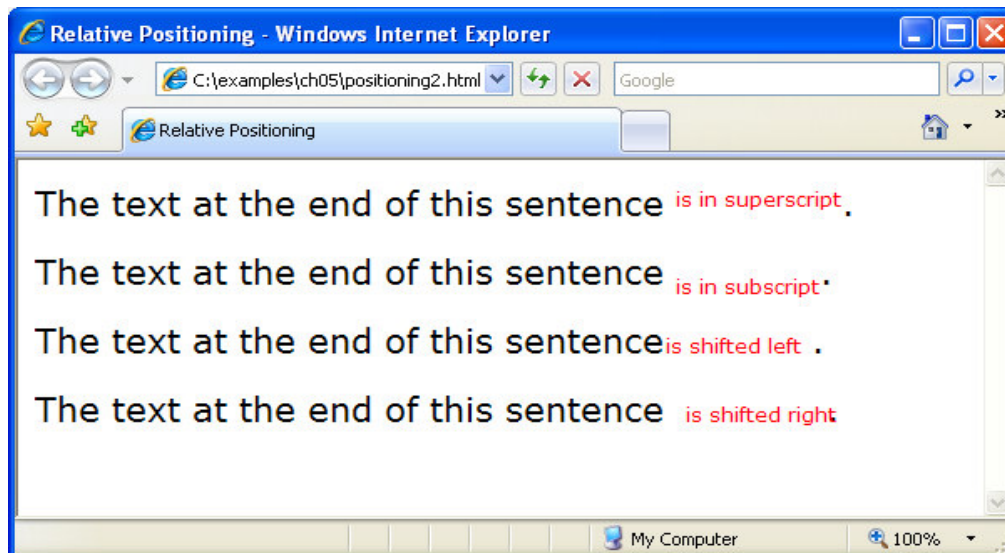
  <p>The text at the end of this sentence
    <span class = "sub">is in subscript</span>.</p>

  <p>The text at the end of this sentence
    <span class = "shiftleft">is shifted left</span>.</p>

  <p>The text at the end of this sentence
    <span class = "shiftright">is shifted right</span>.</p>
</body>
</html>
```

Apply the shiftleft
class to this span element

Apply the shiftright
class to this span element



Backgrounds...

36

- CSS can control the backgrounds of block-level elements by adding:
 - ▣ Colors
 - ▣ Images
- For that, CSS defines 6 properties
 - ▣ background-image
 - ▣ background-position
 - ▣ background-repeat
 - ▣ Background-attachment
 - ▣ Background-color

Backgrounds...

37

- Property `background-image`
 - ▣ Specifies the URL of the image, in the format `url (fileLocation)`
- Property `background-position`
 - ▣ Places the image on the page using the values `top`, `bottom`, `center`, `left` and `right` individually or in combination for vertical and horizontal positioning. You can also position by using lengths

Backgrounds...

38

- `background-repeat` property controls the tiling of the background image
 - ▣ Setting the tiling to `no-repeat` displays one copy of the background image on screen
 - ▣ Setting to `repeat` (the default) tiles the image vertically and horizontally
 - ▣ Setting to `repeat-x` tiles the image only horizontally
 - ▣ Setting to `repeat-y` tile the image only vertically

Backgrounds...

39

- Property setting background-attachment: fixed
 - ▣ fixes the image in the position specified by background-position. Scrolling the browser window will not move the image from its set position. The default value, scroll, moves the image as the user scrolls the window

Backgrounds...

40

- `text-indent` property indents the first line of text in the element by the specified amount
- `font-style` property allows you to set text to `none`, `italic` or `oblique`

Backgrounds

41

```
1 <!DOCTYPE html>
2
3 <html>
4 <head>
5   <meta charset = "utf-8">
6   <title>Background Images</title>
7   <style type = "text/css">
8     body { background-image: url(logo.png);
9            background-position: bottom right;
10           background-repeat: no-repeat;
11           background-attachment: fixed;
12           background-color: LightGray }
13     p { font-size: 18pt;
14         color: DarkBlue;
15         text-indent: 1em;
16         font-family: arial, sans-serif;
17         .dark { font-weight: bold; }
18
19   </style>
20 </head>
21 <body>
22   <p>
23     This example uses the background-image,
24     background-position and background-attachment
25     styles to place the <span class = "dark">Deitel
26     & Associates, Inc.</span> logo in the
27     bottom-right corner of the page. Notice how the logo
28     stays in the proper position when you resize the
29     browser window. The background-color fills in where
30     there is no image.
31   </p>
32 </body>
33 </html>
```

Inserts the image at logo.gif as the background

Places the image at the bottom right of the page

Displays only one copy of the image

Keeps the image in place when the user scrolls in the browser window

Fills the remainder of the window with a light gray background

Indents the first line of text in the element by 1 em



Element Dimensions...

42

- In addition to positioning elements, CSS rules can specify the actual dimensions of each page element
 - ▣ Dimensions of elements on a page can be set with CSS by using properties `height` and `width`
 - Their values can be relative or absolute
- Text in an element can be centered using `text-align: center`; other values for the `text-align` property are `left` and `right`

Element Dimensions...

43

- Problem with setting both vertical and horizontal dimensions of an element
 - ▣ Content might sometimes exceed the set boundaries, in which case the element must be made large enough for all the content to fit
 - ▣ Can set the `overflow` property to `scroll`, which adds scroll bars if the text overflows the boundaries set for it

Element Dimensions...

```
1 <!DOCTYPE html>
2
44 <!-- Fig. 4.12: width.html -->
4 <!-- Element dimensions and text alignment. -->
5 <html>
6 <head>
7   <meta charset = "utf-8">
8   <title>Box Dimensions</title>
9   <style type = "text/css">
10     p { background-color: LightSkyBlue;
11         margin-bottom: .5em;
12         font-family: arial, helvetica, sans-serif; }
13   </style>
14 </head>
15 <body>
16   <p style = "width: 20%;">Here is some
17     text that goes in a box which is
18     set to stretch across twenty percent
19     of the width of the screen.</p>
20
21   <p style = "width: 80%; text-align: center;">
22     Here is some CENTERED text that goes in a box
23     which is set to stretch across eighty percent of
24     the width of the screen.</p>
25
26   <p style = "width: 20%; height: 150px; overflow: scroll;">
27     This box is only twenty percent of
28     the width and has a fixed height.
29     What do we do if it overflows? Set the
30     overflow property to scroll!</p>
31 </body>
32 </html>
33
```

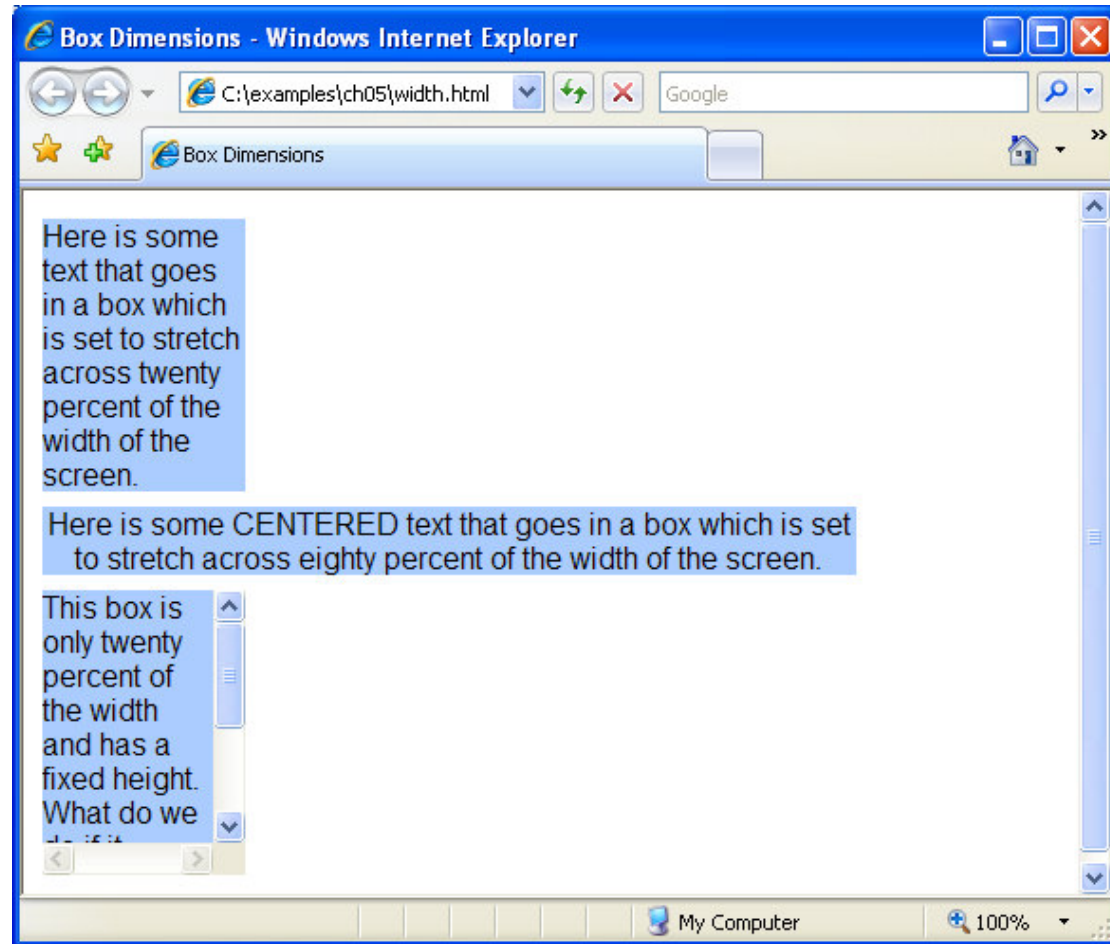
Sets the width of the element to 20% of the browser's screen's size

Sets the width of the element to 80% of the browser's screen's size and centers it

Sets the width of the element to 20% of the browser's screen's size, the height to 150 px, and allows the element to scroll if the text overflows the allotted size

Element Dimension

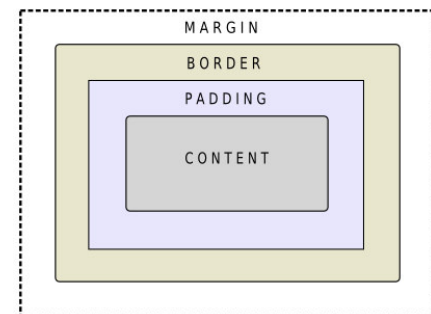
45



Box Model and Text Flow...

46

- Block-level HTML elements have a virtual box drawn around them based on the box model
- When the browser renders using the box model, each element is surrounded by:
 - ▣ Padding
 - The padding property determines the distance between the content inside an element and the edge of the element
 - Padding be set for each side of the box by using `padding-top`, `padding-right`, `padding-left` and `padding-bottom`
 - ▣ Margin
 - Determines the distance between the element's edge and any outside text
 - Margins for individual sides of an element can be specified by using `margin-top`, `margin-right`, `margin-left` and `margin-bottom`
 - ▣ Border



Box Model and Text Flow...

47

- The border is controlled using the properties:
 - ▣ `border-width`
 - May be set to any of the CSS lengths or to the predefined value of `thin`, `medium` or `thick`
 - ▣ `border-color`
 - Sets the color used for the border
 - ▣ `border-style`
 - Options are: `none`, `hidden`, `dotted`, `dashed`, `solid`, `double`, `groove`, `ridge`, `inset` and `outset`

Box Model and Text Flow...

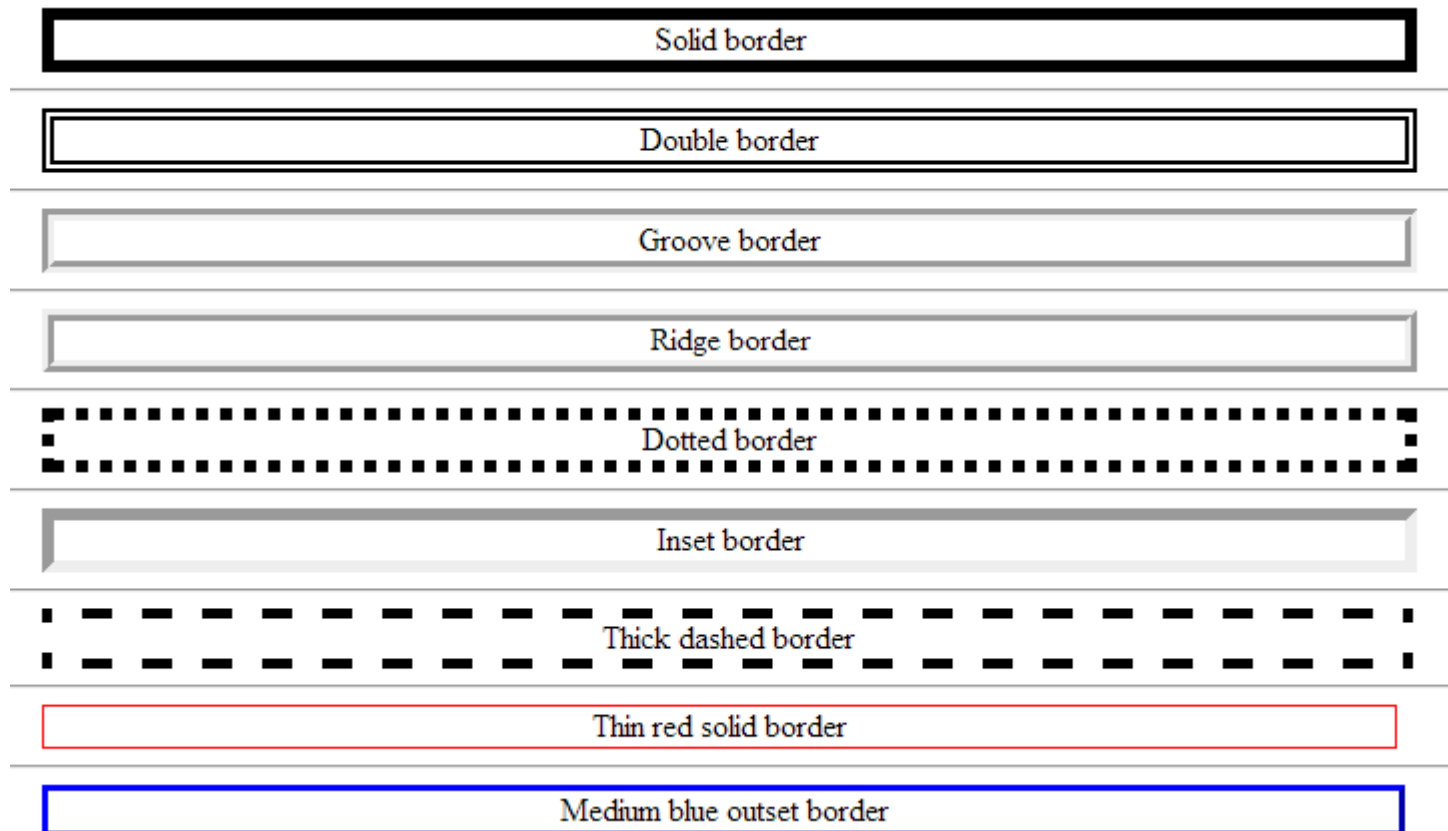
48

- `class` attribute

- ▣ allows more than one class to be assigned to an HTML element by separating each class name from the next with a space

Box Model and Text Flow...

49



Box Model and Text Flow...

50

```
1  <!DOCTYPE html>
2
3  <!-- Fig. 4.14: borders.html -->
4  <!-- Borders of block-level elements. -->
5  <html>
6  <head>
7      <meta charset = "utf-8">
8      <title>Borders</title>
9      <style type = "text/css">
10         div      { text-align: center;
11                   width: 50%;
12                   position: relative;
13                   left: 25%;
14                   border-width: 6px; }
15         .thick   { border-width: thick; }
16         .medium  { border-width: medium; }
17         .thin    { border-width: thin; }
18         .solid   { border-style: solid; }
19         .double  { border-style: double; }
20         .groove  { border-style: groove; }
21         .ridge   { border-style: ridge; }
22         .dotted  { border-style: dotted; }
23         .inset   { border-style: inset; }
24         .outset  { border-style: outset; }
25         .dashed  { border-style: dashed; }
26         .red     { border-color: red; }
27         .blue    { border-color: blue; }
28     </style>
29 </head>
```

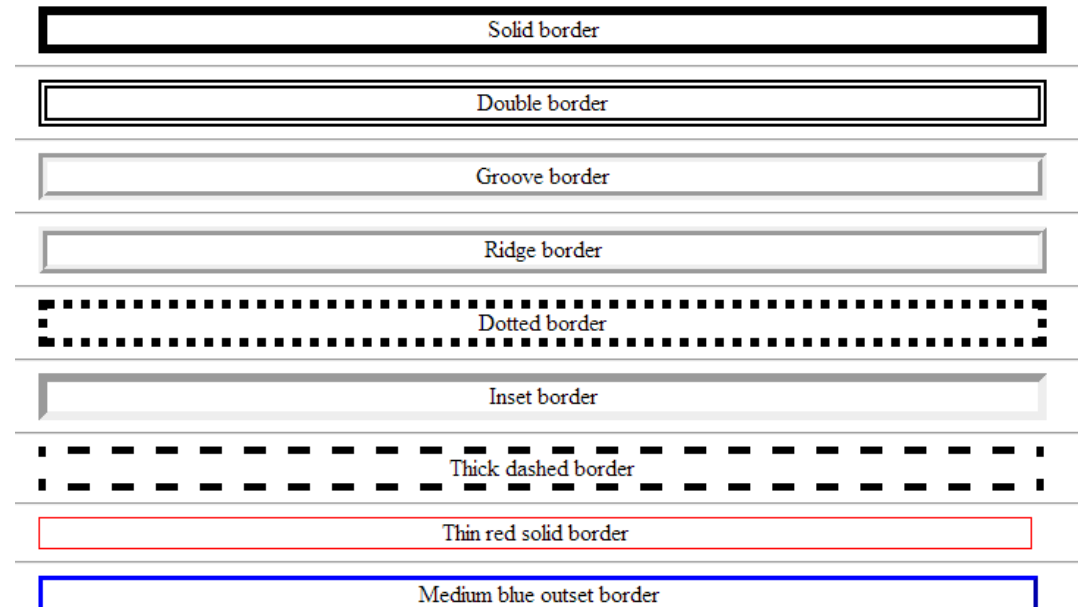
Defines several border classes

Box Model and Text Flow

51

```
<body>
  <div class = "solid">Solid border</div><hr>
  <div class = "double">Double border</div><hr>
  <div class = "groove">Groove border</div><hr>
  <div class = "ridge">Ridge border</div><hr>
  <div class = "dotted">Dotted border</div><hr>
  <div class = "inset">Inset border</div><hr>
  <div class = "thick dashed">Thick dashed border</div><hr>
  <div class = "thin red solid">Thin red solid border</div><hr>
  <div class = "medium blue outset">Medium blue outset border</div>
</body>
</html>
```

Applies several
classes to the same
element



Floating Element...

52

- Browsers normally place text and elements on screen in the order in which they appear in the HTML file
- Elements can be removed from the normal flow of text
- Floating allows you to move an element to one side of the screen; other content in the document will then flow around the floated element

Floating Elements...

53



Floating Elements...

54

```
1 <!DOCTYPE html>
2
3 <!-- Fig. 4.15: floating.html -->
4 <!-- Floating elements. -->
5 <html>
6 <head>
7   <meta charset = "utf-8">
8   <title>Flowing Text Around Floating Elements</title>
9   <style type = "text/css">
10     header { background-color: SkyBlue;
11              text-align: center;
12              font-family: arial, helvetica, sans-serif;
13              padding: .2em; }
14     p { text-align: justify;
15         font-family: verdana, geneva, sans-serif;
16         margin: .5em; }
17     h1 { margin-top: 0px; }
18     .floated { background-color: LightGray;
19               font-size: 1.5em;
20               font-family: arial, helvetica, sans-serif;
21               padding: .2em;
22               margin-left: .5em;
23               margin-bottom: .5em;
24               float: right;
25               text-align: right;
26               width: 50%; }
27     section { border: 1px solid skyBlue; }
28   </style>
29 </head>
```

Sets a spacing of .5 em from the outside of the border to all other content

Sets a spacing of .2 em from the inside of the border to the element's content

Define left and right margins

Moves element to the right, and lets other content flow around it

Defines the border for the section element

Floating Elements

55

```
30 <body>
31   <header><img src = "deitel.png" alt = "Deitel" /></header>
32   <section>
33     <h1 class = "floated">Corporate Training and Authoring</h1>
34     <p>Deitel & Associates, Inc. is an internationally
35       recognized corporate training and authoring organization
36       specializing in programming languages, Internet/web
37       technology, iPhone and Android app development and
38       object technology education. The company provides courses
39       on Java, C++, C#, Visual Basic, C, Internet and web
40       programming, Object Technology and iPhone and Android
41       app development.</p>
42   </section>
43   <section>
44     <h1 class = "floated">Programming Books and Videos</h1>
45     <p>Through its publishing
46       partnership with Pearson, Deitel & Associates,
47       Inc. publishes leading-edge programming textbooks,
48       professional books and interactive web-based and DVD
49       LiveLessons video courses.</p>
50   </section>
51 </body>
52 </html>
53
```

Media Types and Media Queries...

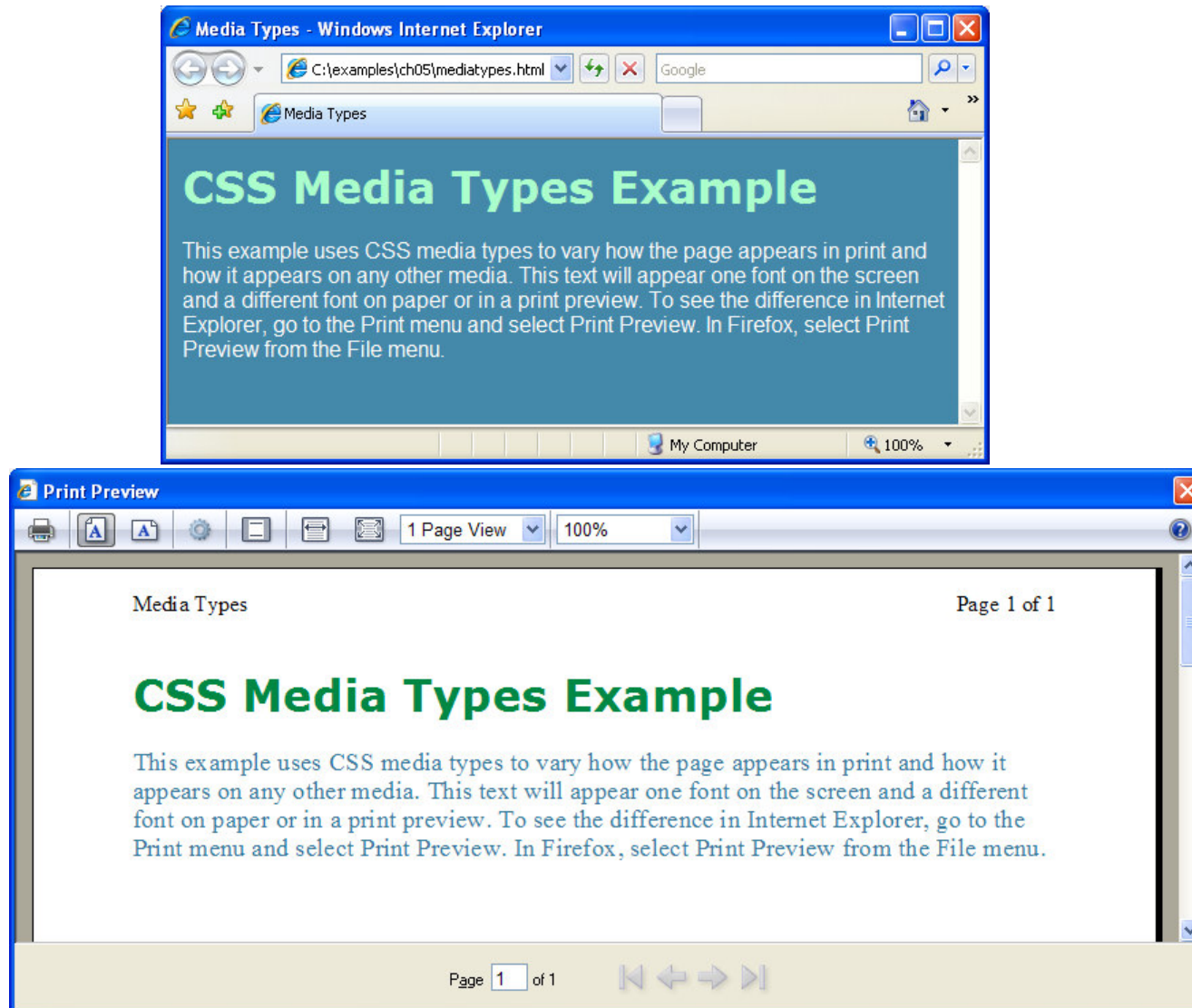
56

□ CSS media types

- ▣ allow a programmer to decide what a page should look like depending on the kind of media being used to display the page
- ▣ Most common media type for a web page is the **Screen** media type, which is a standard computer screen
- ▣ Other media types in CSS include
 - Handheld: For mobile Internet devices
 - Braille: For machines that can read or print web pages in braille
 - Speech: Allow the programmer to give a speech-synthesizing web browser more information about the content of the web page
 - Print: affects a web page's appearance when it's printed

Media Types and Media Queries...

57



Media Types and Media queries...

58

```
1 <!DOCTYPE html>
2
3 <!-- Fig. 4.16: mediatypes.html -->
4 <!-- CSS media types. -->
5 <html>
6 <head>
7   <meta charset = "utf-8">
8   <title>Media Types</title>
9   <style type = "text/css">
10     @media all
11     {
12       body { background-color: SteelBlue; }
13       h1   { font-family: verdana, helvetica, sans-serif;
14             color: PaleGreen; }
15       p    { font-size: 12pt;
16             color: white;
17             font-family: arial, sans-serif; }
18     } /* End @media all declaration. */
19     @media print
20     {
21       body { background-color: white; }
22       h1   { color: SeaGreen; }
23       p    { font-size: 14pt;
24             color: SteelBlue;
25             font-family: "times new roman", times, serif; }
26     } /* End @media print declaration. */
27   </style>
28 </head>
```

Sets properties for all media types

Sets properties for a page if it is being printed

Media Types and Media queries...

59

```
29 <body>
30   <h1>CSS Media Types Example</h1>
31
32   <p>
33     This example uses CSS media types to vary how the page
34     appears in print and how it appears on any other media.
35     This text will appear in one font on the screen and a
36     different font on paper or in a print preview. To see
37     the difference in Internet Explorer, go to the Print
38     menu and select Print Preview. In Firefox, select Print
39     Preview from the File menu.
40   </p>
41 </body>
42 </html>
```

Look-and-Feel Observation 1

Pages with dark background colors and light text use a lot of ink and may be difficult to read when printed, especially on a black-and white-printer. Use the print media type to avoid this.

Look-and-Feel Observation 2

In general, sans-serif fonts look better on a screen, while serif fonts look better on paper. The `print` media type allows your web page to display sans-serif font on a screen and change to a serif font when it is printed.

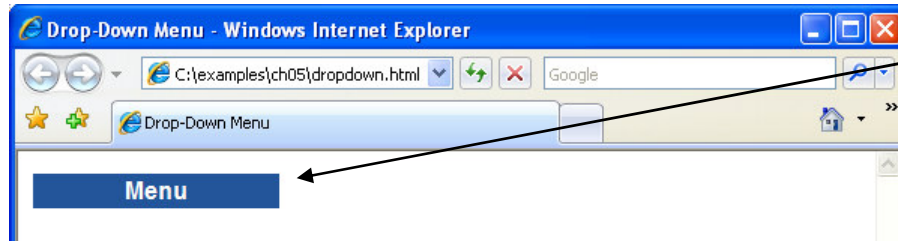
Drop-Down Menu...

62

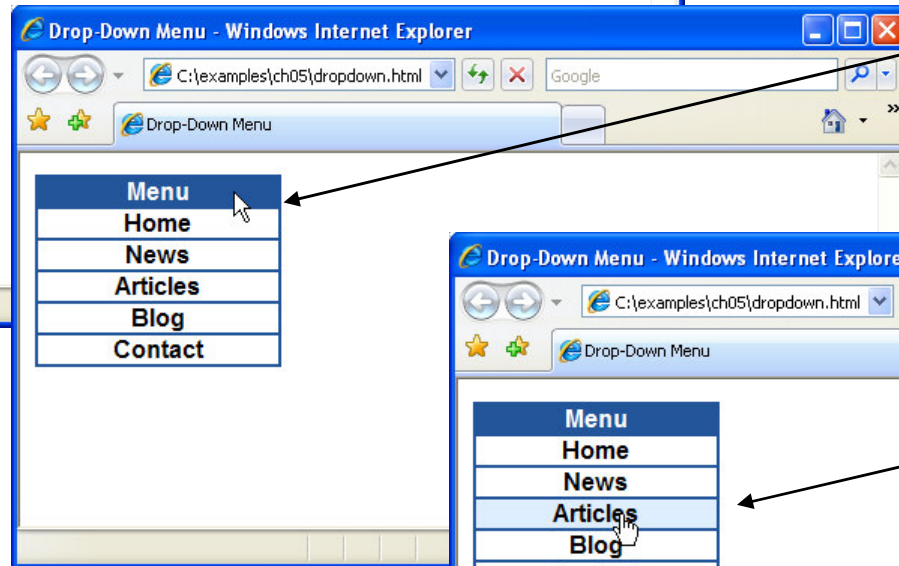
- We have seen
 - ▣ the `:hover` pseudoclass used to change a link's style when the mouse hovers it
 - ▣ The `display` property allows a programmer to decide if an element is displayed as a block element, inline element, or is not rendered at all (`none`)
- We can use also `:hover` to cause a menu to appear when the mouse hovers over a menu button and `display` property which allows you to decide whether an element is rendered on the page or not

Drop-Down Menu...

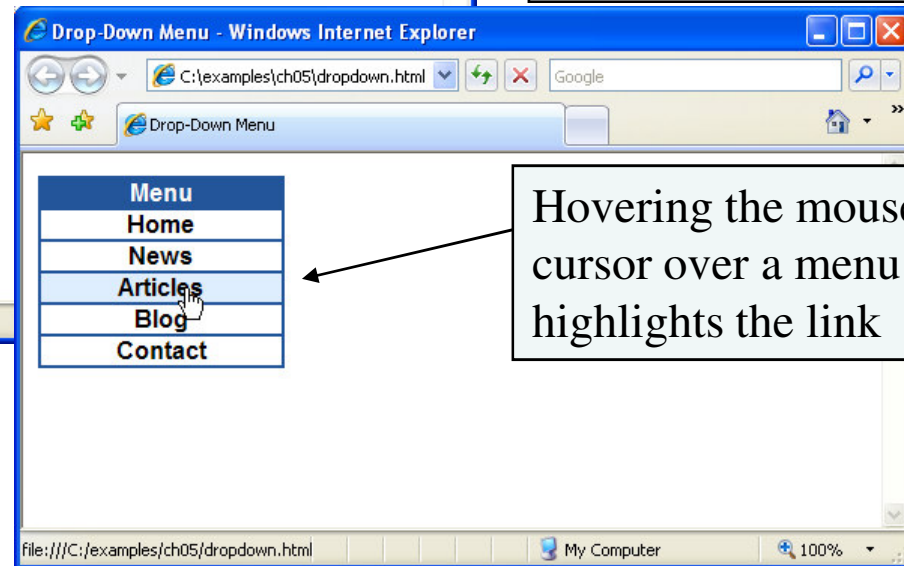
63



A collapsed menu



A drop-down menu is displayed when the mouse cursor is hovered over Menu



Hovering the mouse cursor over a menu link highlights the link

Drop-Down Menu...

64

```
1 <!DOCTYPE html>
2
3 <html>
4   <head>
5     <meta charset = "utf-8">
6     <title>
7       Drop-Down Menu
8     </title>
9     <style type = "text/css">
10      body
11      nav
12      { font-family: arial, sans-serif }
13      { font-weight: bold;
14        color: white;
15        border: 2px solid RoyalBlue;
16        text-align: center;
17        width: 10em;
18        background-color: RoyalBlue; }
19
20      nav ul
21      { display: none;
22        list-style: none;
23        margin: 0;
24        padding: 0; }
25
26      nav:hover ul
27      { display: block; }
28
29      nav ul li
30      { border-top: 2px solid RoyalBlue;
31        background-color: white;
32        width: 10em;
33        color: black; }
34
35      nav ul li:hover
36      { background-color: PowderBlue; }
37
38      a
39      { text-decoration: none; }
40    </style>
41  </head>
```

Sets display to none for any unordred list that's nested in a nav

Selects only ul elements nested in a nav element that currently has the mouse hovering over it

This style is applied only to li element that's child of a ul element in a nav element and only when that li has the mouse cursor over it

Drop-Down Menu

65

```
<body>
  <nav>Menu
    <ul>
      <li><a href = "#">Home</a></li>
      <li><a href = "#">News</a></li>
      <li><a href = "#">Articles</a></li>
      <li><a href = "#">Blog</a></li>
      <li><a href = "#">Contact</a></li>
    </ul>
  </nav>
</body>
</html>
```

Create a nav Element containing the text Menu and an unordred list of 5 links that should appear in the drop-down menu

Conflicting Styles

66

- CSS Style Sheets can be defined by
 - ▣ The author (Inline, Embedded and External Style Sheets)
 - ▣ The user
 - ▣ The agent User (browser). All browsers have a default Style Sheet called: **"user agent style sheet"**
- Styles defined by authors take precedence over styles defined by the user
- Styles defined by the user take precedence over styles defined by the user agent
- Most styles are inherited from parent elements. Styles defined for children have higher specificity and take precedence over the parent's styles
- Conflicts are resolved in favor of properties with a higher specificity

References

67

- TextBook and Pearson slides
- Wikipedia

