

HTML5 and CSS3

Introductory

Second Edition

Unit D

Laying out Elements with CSS

Unit D

Review id vs class



Objectives

- Assess the CSS box model
- Set element width and borders
- Set margins and padding
- Align elements with float
- Control page flow with clear

Objectives (continued)

- Implement fixed positioning
- Implement relative positioning
- Control stacking order
- Implement absolute positioning

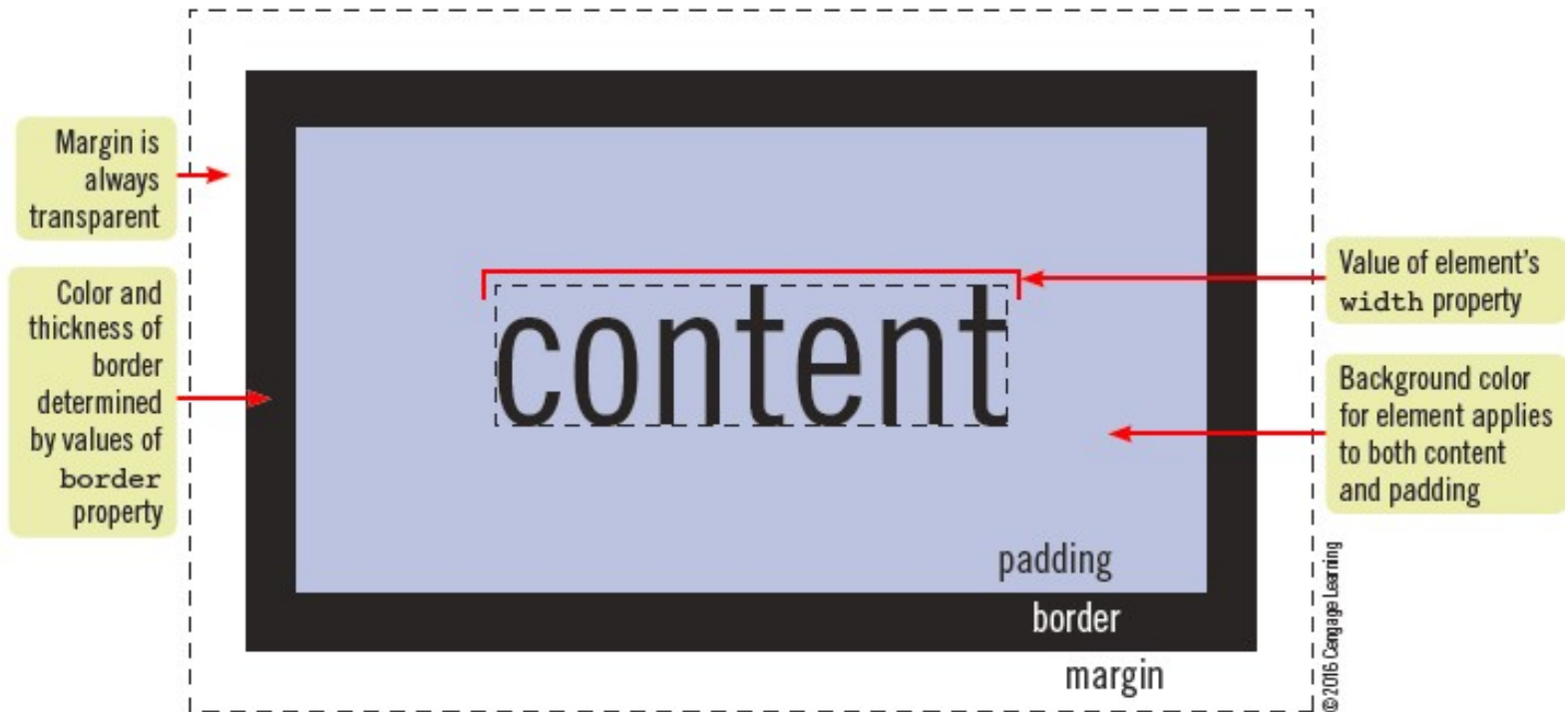
Assess the CSS Box Model

- Box model: used by CSS to represent characteristics of every web page element
 - Treats element as rectangular box
- border: border surrounding element
- margin: space from border to neighboring/parent element
- padding: space between border and element content

Unit D

Assess the CSS Box Model (continued)

- The CSS box model



Unit D

Assess the CSS Box Model (continued)

- Box model properties

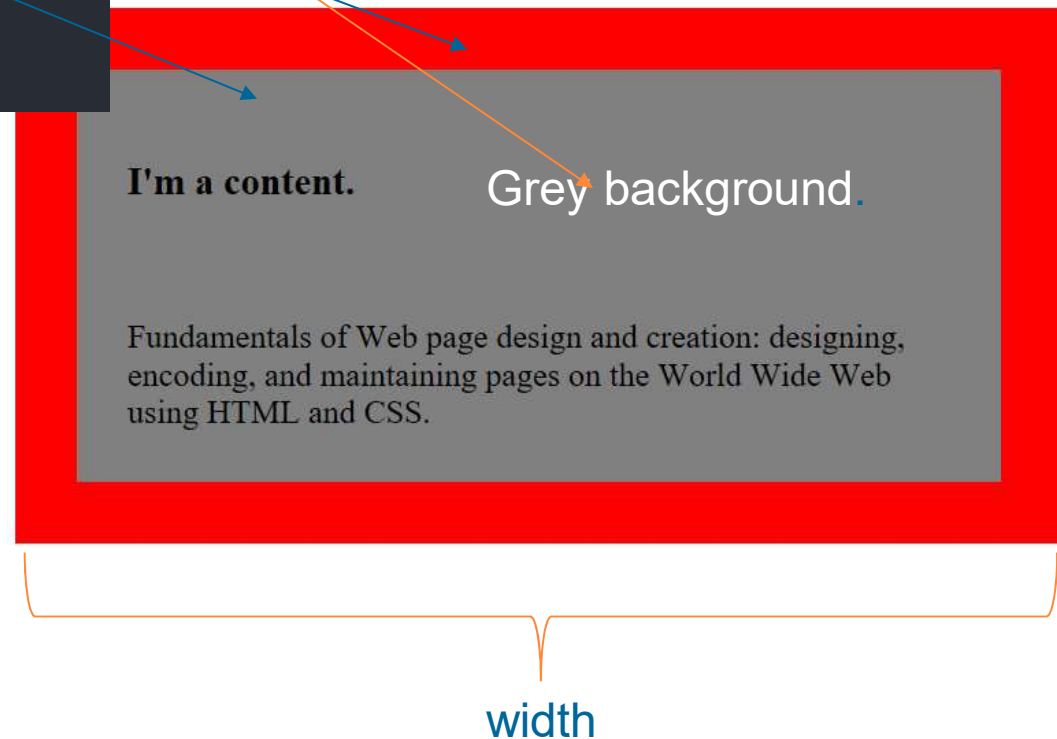
property	description	example*
<code>border</code>	a visible border around an element specified with a thickness value, a style keyword, and a color name or value	<code>border: 1px solid black;</code>
<code>padding</code>	the space inside a border between the border and the element content	<code>padding: 5%;</code>
<code>margin</code>	the space outside a border between the border and adjacent or parent elements	<code>margin: 10px;</code>
<code>width</code>	the horizontal size of an element's content, excluding border, padding, and margin	<code>width: 25%;</code>
<code>min-width</code>	the minimum horizontal size of an element's content, excluding border, padding, and margin	<code>min-width: 320px;</code>
<code>max-width</code>	the maximum horizontal size of an element's content, excluding border, padding, and margin	<code>max-width: 1000px;</code>
<code>height</code>	the vertical size of an element's content, excluding border, padding, and margin	<code>height: 5rem;</code>
<code>min-height</code>	the minimum vertical size of an element's content, excluding border, padding, and margin	<code>min-height: 2em;</code>
<code>max-height</code>	the maximum vertical size of an element's content, excluding border, padding, and margin	<code>max-height: 5em;</code>

Unit D

Demo Box Model

boxmodel_demo.html
Using <div> tag.

```
<style>
div {
  background-color: grey;
  width: 400px;
  border: 30px solid red;
  padding: 25px;
  margin: 25px;
}
</style>
```



Unit D

Assess the CSS Box Model (continued)

- CSS units of measure
 - Relative units: ems, percent, rem
 - Absolute units: pixels
- Commonly used CSS units of

abbreviation	unit	description	example
em	em	A multiple of the computed font size for the current element, where 1em represents 100% of this size	padding: 1.4em;
%	percent	A percentage of another value for the current element or an ancestor element; each CSS property that takes a value in % specifies what value the % calculation is based on	margin: 10%;
px	pixel	A unit equal to approximately 1/96 inch or 0.26 millimeter	padding: 10px;
rem	rem	A multiple of the computed font size for the html element, where 1rem represents 100% of this size	margin: 0.25rem;

Unit D

CSS Units of measure Demo

```
<style>
div {
  background-color: grey;
  width: 30%;
  border: 30px solid yellow;
  padding: 3em;
  margin: 10%;
}

p {
  font-size: 30px;
}
</style>
```

css_units_demo.html

1. 3em means 2 x size of
The current font.

Box Model (CSS Units - em, percent, pixel, rem)

Demo by Prof. Castro 30px

I'm a content.

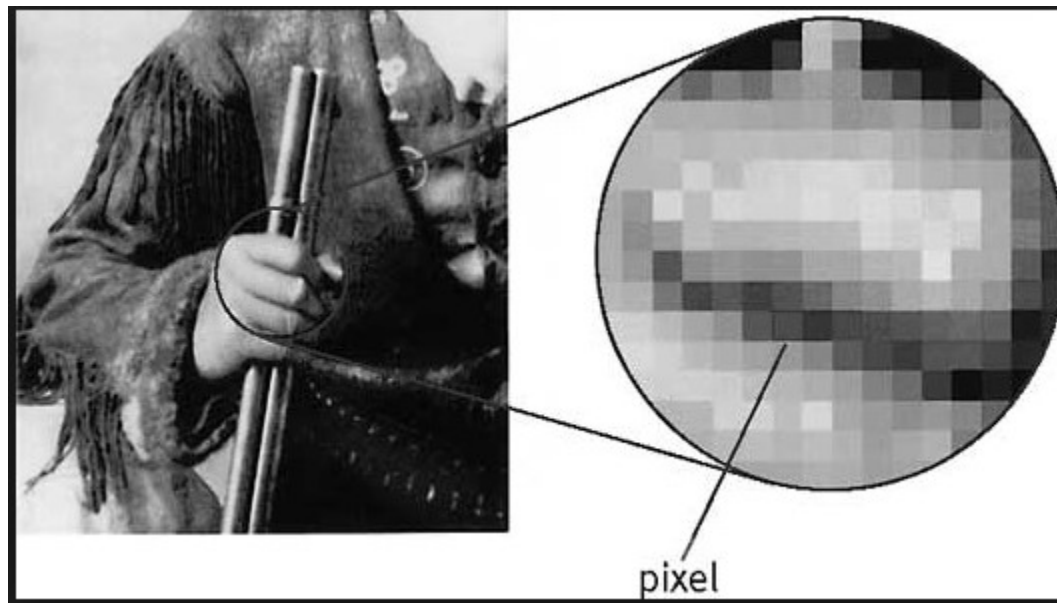
Fundamentals of Web page design and creation: designing, encoding, and maintaining pages on the World Wide Web using HTML and CSS.

Unit D

What is pixel?

Dictionary.com:

The smallest element of an image that can be individually processed in a video display system.



If you're watching TV, it has thousands of pixels.

Unit D

Sometimes you get dead pixel.



Resolutions	Pixels on Screen
1366 x 768	1,049,088
1920 x 1080	2,073,600

If you see a black dot on your TV. That's a dead pixel.

Assess the CSS Box Model (continued)

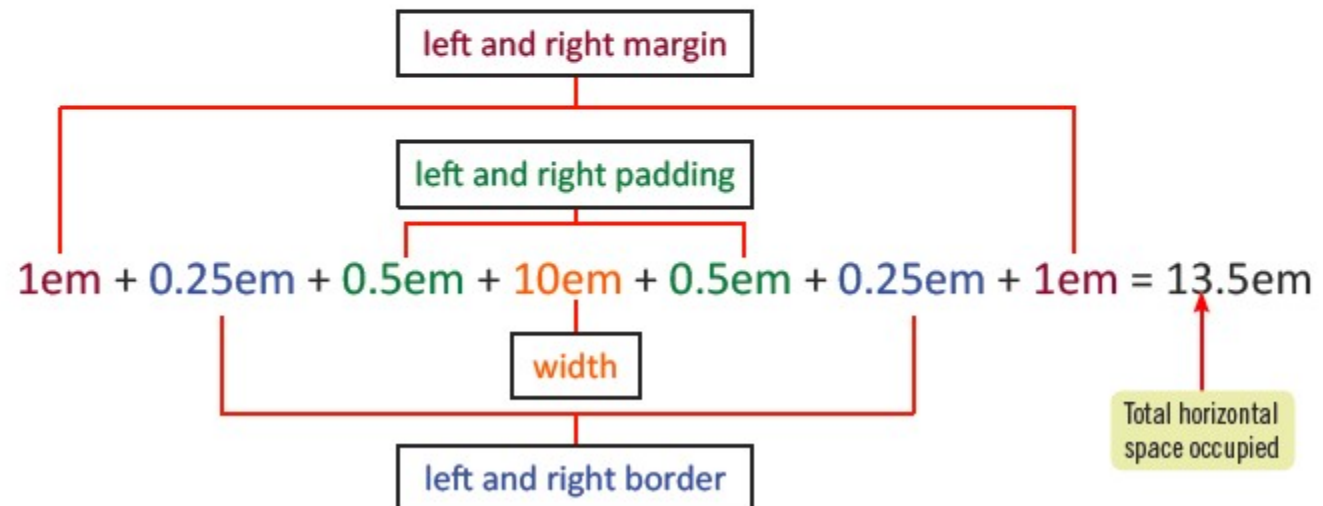
- Size of `padding`, `margin`, and `border` increase the amount of space occupied by an element
 - Dimensions of these properties not included in specified width / height
 - Specified width and height refer only to the content of an element
 - To calculate add the total space, add the `width` value to the `padding`, `margin`, and `border` values

Unit D

Assess the CSS Box Model (continued)

- Calculating web page space occupied by an element

```
p {  
  width: 10em;  
  padding: 0.5em;  
  border: 0.25em solid black;  
  margin: 1em;  
}
```



Unit D

Calculating web space occupied demo

boxmodel_calculate.html

```
div {  
    font-size: 30px;  
    background-color: grey;  
    width: 10em;  
    border: 1em solid black;  
    padding: 0.5em;  
    margin: 1em;  
}
```

Demo by Prof. Castro 10px

I'm a content.

Fundamentals of Web
page design and
creation: designing,
encoding, and
maintaining pages on
the World Wide Web
using HTML and CSS.

Unit D



Assess the CSS Box Model (continued)

- Collapse: When bottom margin of one element is adjacent to top margin of another, the margins combine to the size of the larger of the two
 - Affects element size planning

Unit D

Assess the CSS Box Model

Collapsing Margins demo between adjacent elements.

[collapsing_margin_demo.html](#)

In the example above, the h1 element has a bottom margin of 50px. The h2 element has a top margin set to 10px. Looking at it, it would appear that the vertical margin between the h1 and the h2 would be a total of 30px (10px + 20px). Because of the margin collapse, the actual margin ends up being 10px.

NOTE: Margin is invisible.

Fundamentals of Web page design and creation (top)

Fundamentals of Web page design and creation (bottom)

Collapse in between.

Set Element Width and Borders

- Margin, padding, and border are shorthand properties
 - Assign values to multiple CSS properties
 - Assign values to all four sides of an element
 - Assign these values using a single declaration
- Can set width, style, or color of any or all sides of an element

Unit D

Set Element Width and Borders (continued)

- Specific properties for the box model

component	properties	description	values
border	border-width-side	thickness of a visible border	a border thickness, such as 1px
	border-style-side	style of a visible border	dashed, solid, or another keyword
	border-color-side	color of a visible border	a color name or value, such as black
padding	padding-side	space between border and content	a value in a CSS unit, such as 5%
margin	margin-side	space between border and adjacent elements	a value in a CSS unit, such as 1.2em

- Reset rule: resets one or more common properties to a common baseline, e.g.

➤ `border: 0`

Unit D

Set Element Width and Borders (continued)

- Code with `width` and `border` declarations

```
10  /* reset styles */
11  article, body, div, footer, header, h1, h2, p {
12      border: 0;
13  }
14
15  /* body and page container */
16  .container {
17      max-width: 640px;
18      background-color: beige;
19  }
```

Using `max-width` allows content to be displayed at a narrower width on smaller screens; note that declarations for box model properties are listed first within a style rule by convention

```
30  /* sidebar */
31  aside {
32      background-color: goldenrod;
33  }
34  aside p {
35      border-bottom: 1px solid black;
36      border-top: 1px solid black;
37  }
```

Declarations add top and bottom borders to `aside p` element without adding left or right borders

These are specific properties for components of the box model

Unit D

Set Element Width and Borders (continued)

- width and border declarations rendered in a browser and developer tools opened to show box model

Maximum width of element with container class value set to 640px

Lakeland Reeds
Bed & Breakfast

About Us

Lakeland Reeds is an ideal place to unspool from your daily routine, whether on the shore of Twin Lakes watching the sunset over the cattails, or just curled up in a hammock on our wraparound porch with a good book. Breakfast is ready whenever you are, and we're happy to accommodate a wide range of appetites, from comfort food, like sausage and biscuits, to the lighter end of things, such as egg white omelettes and fresh fruit.

We're here to help you get the most out of your time with us, from canoes and life vests for a day out on the lake, to popcorn and a DVD of The Wizard of Oz for a quiet evening in. We look forward to welcoming you!

What a weekend! Lakeland Reeds was the perfect place to get away from the hubbub of the city. I'll be back as often as I can get away.

Phillip Blaine, Proprietor

45 Marsh Grass Ln. • Marble, MN 55764 • (218) 555-5253

br element adds a line break within the h1 element, splitting the content into two lines

1px black border added to top and bottom of p element within aside element

Illustration of box model values in Chrome developer tools

p element within aside element selected in browser tools

Browser default value for margin; your browser may also add a default value for padding

```
<!DOCTYPE html>
<html lang="en">
<head>
</head>
<body>
  <div class="container">
    <header>
      <h1>Lakeland Reeds Bed & Breakfast</h1>
    </header>
    <article>
      <h2>About Us</h2>
      <p>Lakeland Reeds is an ideal place to unspool from your daily routine, whether on the shore of Twin Lakes watching the sunset over the cattails, or just curled up in a hammock on our wraparound porch with a good book. Breakfast is ready whenever you are, and we're happy to accommodate a wide range of appetites, from comfort food, like sausage and biscuits, to the lighter end of things, such as egg white omelettes and fresh fruit.</p>
      <p>We're here to help you get the most out of your time with us, from canoes and life vests for a day out on the lake, to popcorn and a DVD of The Wizard of Oz for a quiet evening in. We look forward to welcoming you!</p>
      <p>What a weekend! Lakeland Reeds was the perfect place to get away from the hubbub of the city. I'll be back as often as I can get away.</p>
      <div>
        <p>Phillip Blaine, Proprietor</p>
        <p>45 Marsh Grass Ln. • Marble, MN 55764 • (218) 555-5253</p>
      </div>
    </article>
  </div>
</body>
</html>
```

margin: 10px
border: 1px solid black
padding: 10px
width: 300px

Unit D

DEMO

element_width_borders.html

```
div {  
  background-color: grey;  
  width: 30%;  
  border: 30px solid yellow;  
  padding: 3em;  
  margin: 4%;  
  border-top-width: thin;  
  border-right-color: red;  
}
```

There are many properties you can use to manipulate the elements. You can use atom code editor to automatically pop-pop the properties when you're coding.

I'm a content.

Fundamentals of Web page design and creation: designing, encoding, and maintaining pages on the World Wide Web using HTML and CSS.

Set Margins and Padding

- Use the shorthand property to assign multiple values
 - Separate each value with a space
 - Meaning of different number of values

# of values	values apply to	example
four	top, right, bottom, left	<code>margin: 0 5% 5% 0;</code>
three	top, left/right, bottom	<code>padding: 0 2em 1em;</code>
two	top/bottom, left/right	<code>margin: 0 10px;</code>
one (shorthand property)	top/left/bottom/right	<code>padding: 5%;</code>

- Assign `auto` to left and right margins to center the element horizontally

Set Margins and Padding (continued)

- Example of `padding` using three values and `margin` using two values

```
47  aside p {  
48      border-bottom: 1px solid black;  
49      border-top: 1px solid black;  
50      padding: 0.4em 0;  
51      margin: 0 0.6em;  
52  }
```

- Example of `margin` set to `auto`

```
17  /* body and page container */  
18  .container {  
19      max-width: 640px;  
20      margin: 0 auto;  
21      background-color: beige;  
22  }
```

Unit D

Demo (Short Hand Properties)

short_hand_demo.html

```
div {  
  border: 1px solid red;  
  margin: 50px 200px 200px 300px;  
  background-color: green;  
}
```

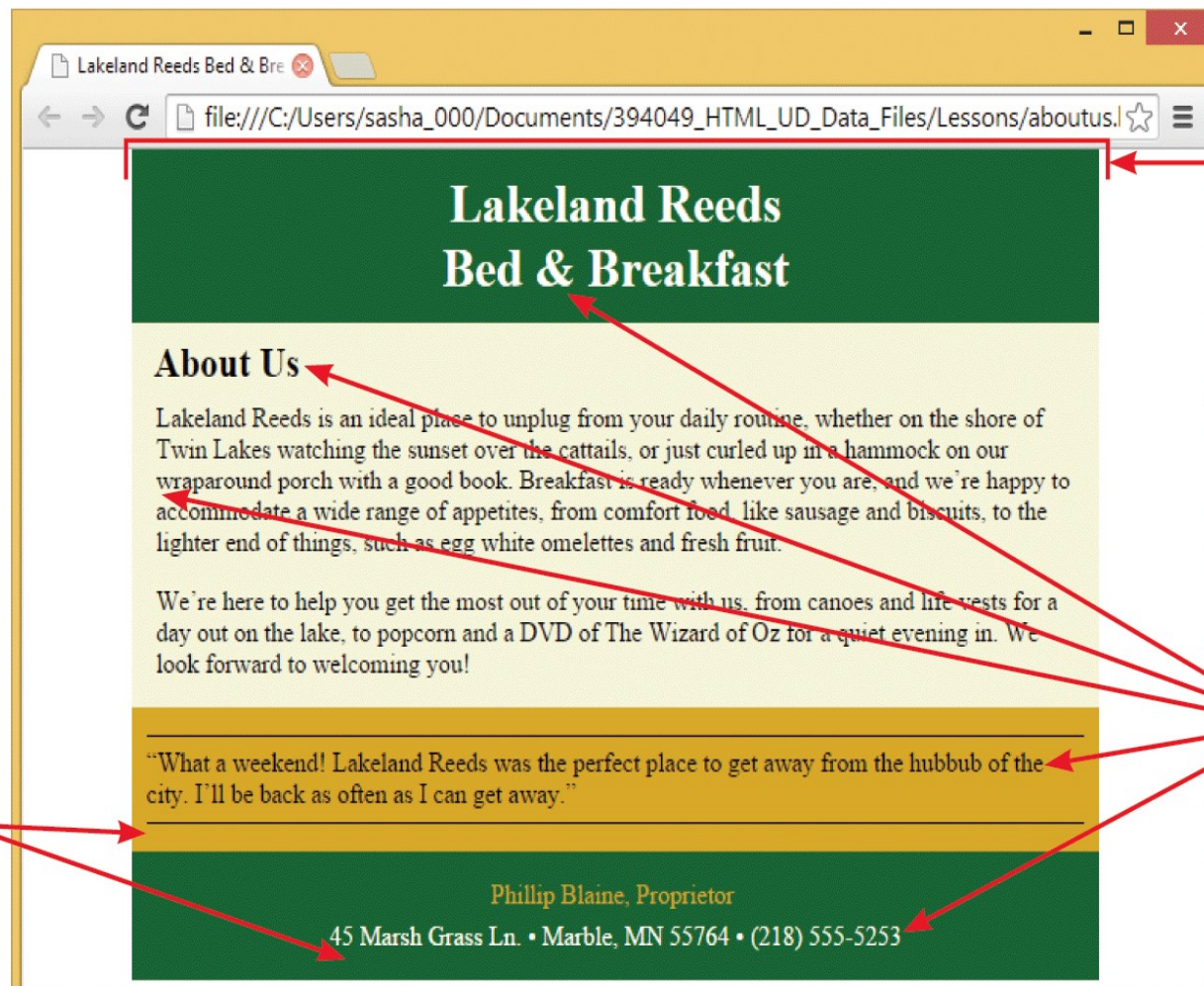
Demo of "Short Hand Properties" by Prof. Castro

1. 50px top margin
2. 200px of right margin
3. 200px of bottom margin
4. 300px of left margin

Fundamentals of Web page design and creation: designing, encoding,
and maintaining pages on the World Wide Web using HTML and CSS.

Unit D

Classroom Activities



Element with container class value element centered in browser window because left and right margins set to auto

Padding added to h1, h2, aside, and footer elements as well as to paragraphs in article and aside elements

Margins added to paragraphs in aside and footer elements

Unit D

D4 - TAbble

# of values	values apply to	example
four	top, right, bottom, left	<code>margin: 0 5% 5% 0;</code>
three	top, left/right, bottom	<code>padding: 0 2em 1em;</code>
two	top/bottom, left/right	<code>margin: 0 10px;</code>
one (shorthand property)	top/left/bottom/right	<code>padding: 5%;</code>

Unit D

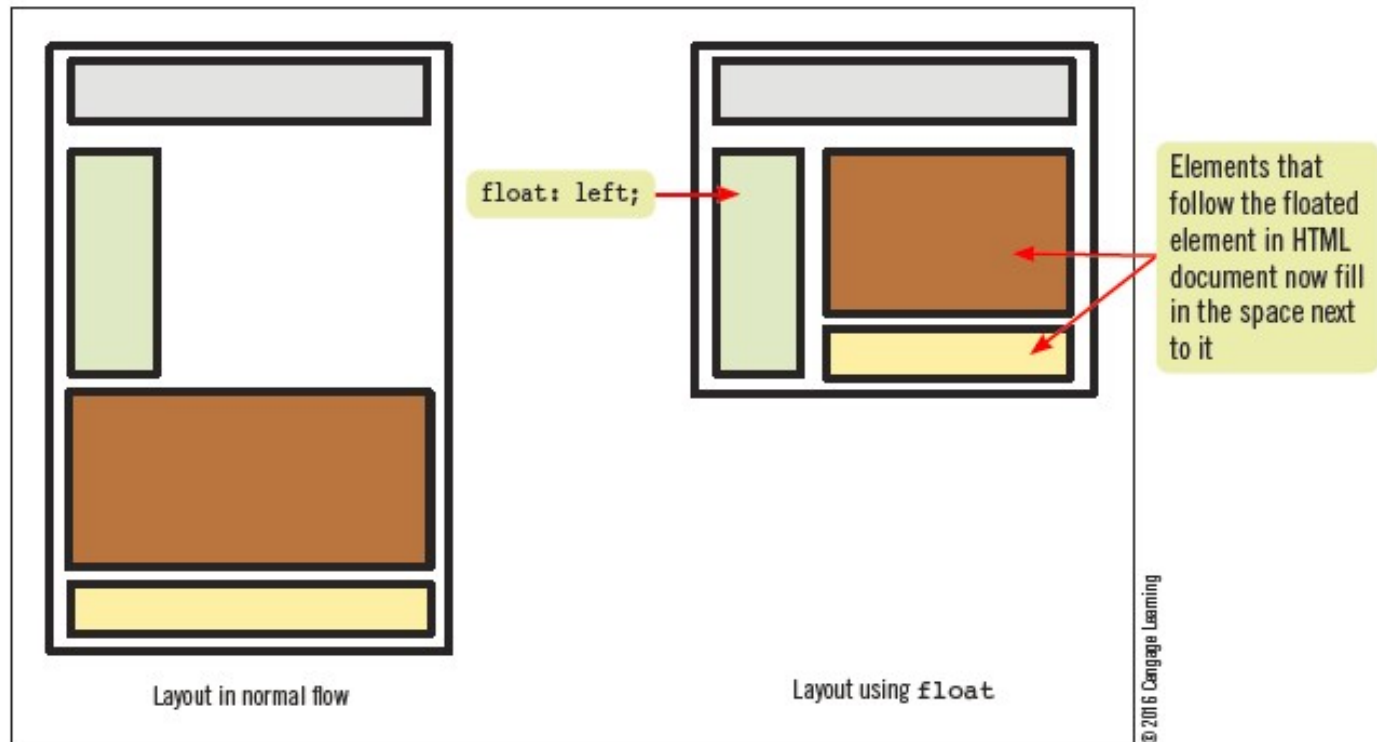
Align Elements with `float`

- Page flow: the order of elements in the HTML document
 - User agents render HTML top to bottom
- Use `float`, `clear`, and `width` properties to create columns of text and graphics parallel to each other
 - Use `width` property to assign width to each of the columns

Unit D

Align Elements with `float` (continued)

- Example of `float` applied to element



- Use `float` to create multicolumn layouts

Unit D

Align Elements with `float` (continued)

- Code with `float` and result in browser

```
37  /* main content */
38  article {
39      width: 70%;
40      float: right;
41  }
42  article p {
43      padding: 0 1em 1em;
44  }
```

```
45
46  /* sidebar */
47  aside {
48      width: 30%;
49      padding: 1em 0;
50      background-color: goldenrod;
51  }
```



Unit D

Demo

css_float_demo.html



```
img {  
  float: left;  
  max-width: 160px;  
  margin: 0;  
  padding: 1em;  
}
```

```
nav {  
  float: left;  
  max-width: 160px;  
  margin: 0;  
  padding: 1em;  
}
```

```
article {  
  margin-left: 170px;  
  border-left: 1px solid gray;  
  padding: 1em;  
  overflow: scroll; /*This property  
  float: none;  
}
```

Unit D

Classroom Activities

Each of the layouts can be achieved using tables. Do you think you can achieve the same layout using floating elements.

Yep

Table is not just easy to use



Control Page Flow with `clear`

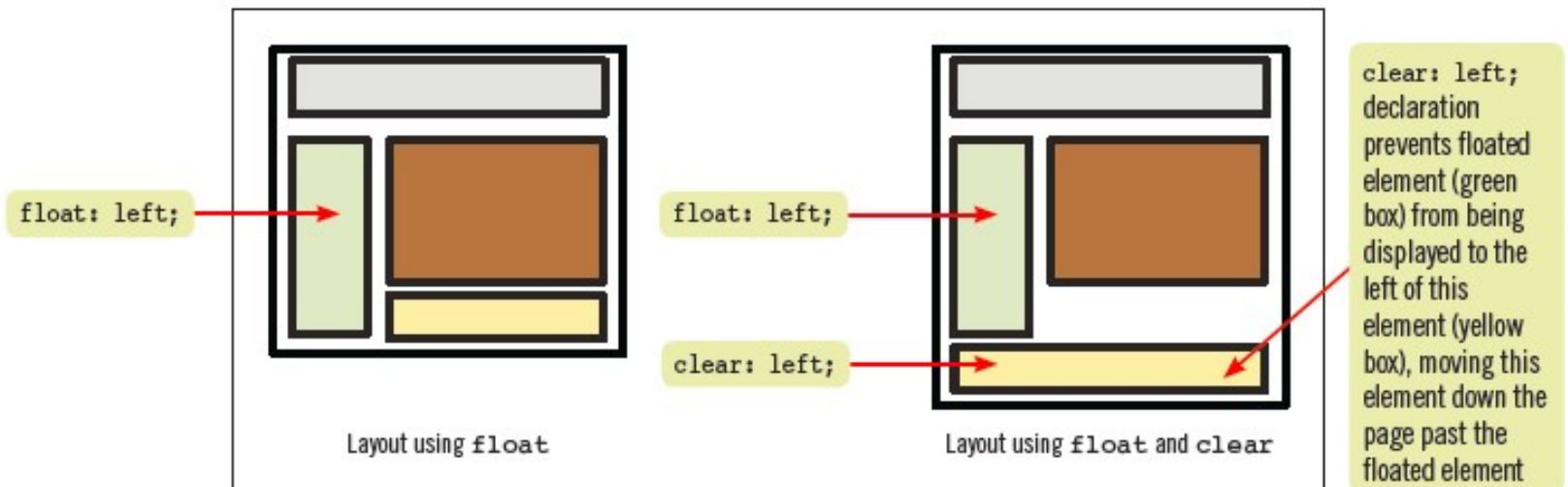
- `float` gives basic control over layout
- `clear` gives more precise control
 - Prevents floated elements from being displayed to the left, right, or on the side of another element
- `clear` property values

value	description
<code>left</code>	element is displayed at next location in the document where no element is floated to the left
<code>right</code>	element is displayed at next location in the document where no element is floated to the right
<code>both</code>	element is displayed at next location in the document where no element is floated either to the left or to the right
<code>none</code>	element is displayed at next available location in the document, regardless of surrounding floated elements

Unit D

Control Page Flow with `clear` (continued)

- Example of `clear` property



Unit D

Control Page Flow with `clear` (continued)

- Code including `clear` property and how it is rendered in a browser

```
59  /* footer section */
60  footer {
61    padding: 0.6em;
62    color: white;
63    background-color: darkgreen;
64    text-align: center;
65    clear: right;
66  }
```

Footer element
displayed in
normal flow after
right margin is
clear of floated
elements



Unit D

Demo

Control Page with clear css_clear_layout.html

```
/* footer section */  
footer {  
  border-top: 3px solid red;  
  padding: 0.5em;  
  clear: both;  
  background-color: yellow;  
  text-align: center;  
}
```

Open Game Store Shop

For all gamers

<h3>Summer Hours</h3> <p>Mon: 8am — 8pm Tue: 8am — 8pm Wed: 8am — 5pm Thu: 8am — 8pm Fri: 8am — 8pm Sat: 9am — 6pm</p>	<h3>Specials</h3> <h4>Annuals</h4> <p>50% Off All games..</p> <h4>Summer</h4> <p>\$1 per game 25% Off All. 20% Off younger than 20.</p>
--	---

548 N. 58th St. • Monterey, CA 98132 • [\(402\) 555-9736](tel:(402)555-9736)

Unit D

Demo Control Page with clear

```
/* footer section */  
footer {  
  border-top: 3px solid red;  
  padding: 0.5em;  
  background-color: yellow;  
  text-align: center;  
}
```

Clear was
deleted.

Open Game Store Shop

For all gamers

<h4>Summer Hours</h4> <p>Mon: 8am — 8pm Tue: 8am — 8pm Wed: 8am — 5pm Thu: 8am — 8pm Fri: 8am — 8pm Sat: 9am — 6pm</p>	<h4>Specials</h4> <h5>Annals</h5> <p>50% Off All games..</p> <h5>Summer</h5> <p>\$1 per game 25% Off All. 20% Off younger than 20.</p>
--	--

548 N. 58th St. • Monterey, CA 98132 • [\(402\) 555-9736](tel:4025559736)

Implement Fixed Positioning

- `fixed` positioning keeps an element in the same location, even when the page is scrolled
- Use the `position` property with the value of `fixed`
- Then specify
 - Horizontal position using `left` or `right`
 - Vertical positing using `top` or `bottom`

Unit D

Implement Fixed Positioning (continued)

- Commonly used position properties

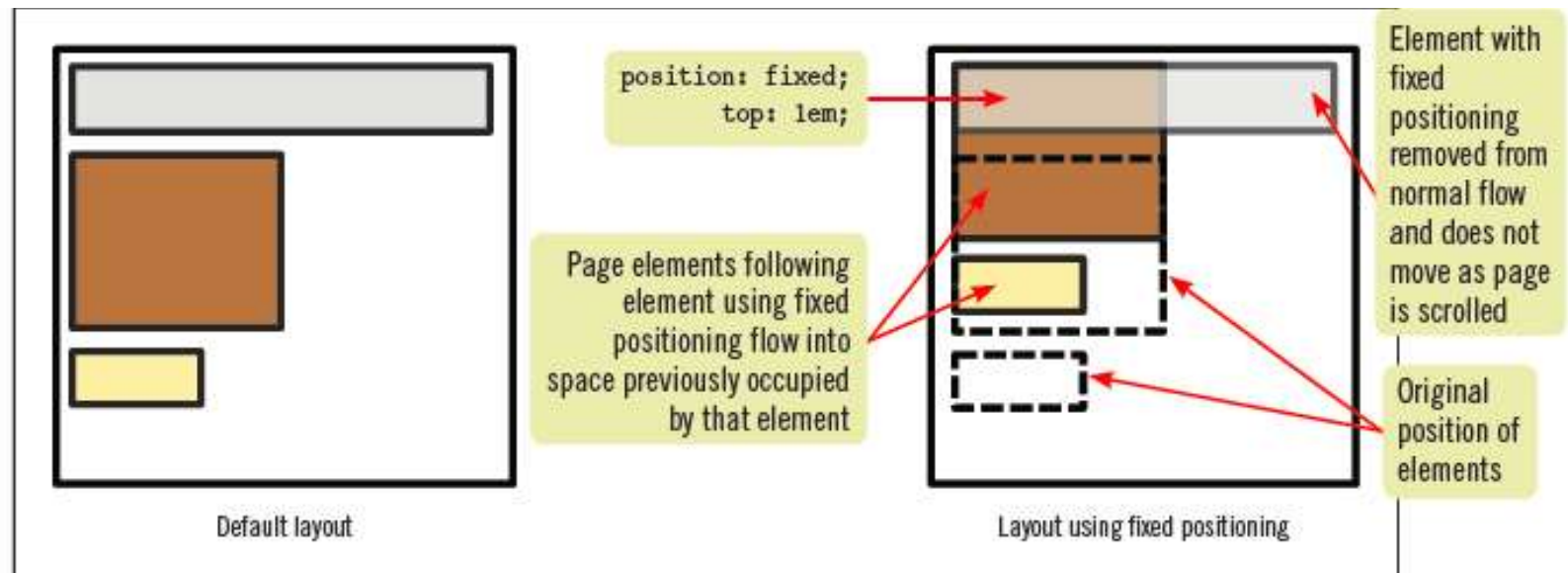
value	description
<code>absolute</code>	element is removed from the normal flow and positioned relative to the nearest ancestor element that has a position value other than <code>static</code>
<code>fixed</code>	element is removed from the normal flow and positioned relative to the browser window, and remains in this position even as a user scrolls through the document
<code>relative</code>	element remains in the normal flow and is positioned relative to its default position
<code>static</code> (default)	element is displayed in its default position in the normal flow

- Properties and values for `fixed` positioning

property	value	description
<code>position</code>	<code>fixed</code>	Removes element from normal flow, positions it relative to the browser window, and keeps it in this position even as a user scrolls through the document
<code>top</code> , <code>bottom</code>	value in ems, pixels, or another supported unit, or percent of the height of the browser window	Moves the element the specified distance down from the top edge (<code>top</code>) or up from the bottom edge (<code>bottom</code>) of the browser window; the default value, 0, leaves the element in its original vertical position
<code>left</code> , <code>right</code>	value in ems, pixels, or another supported unit, or percent of the width of the browser window	Moves the element the specified distance right from the left edge (<code>left</code>) or left from the right edge (<code>right</code>) of the browser window; the default value, 0, leaves the element in its original horizontal position

Unit D

Implement Fixed Positioning (continued)



Unit D

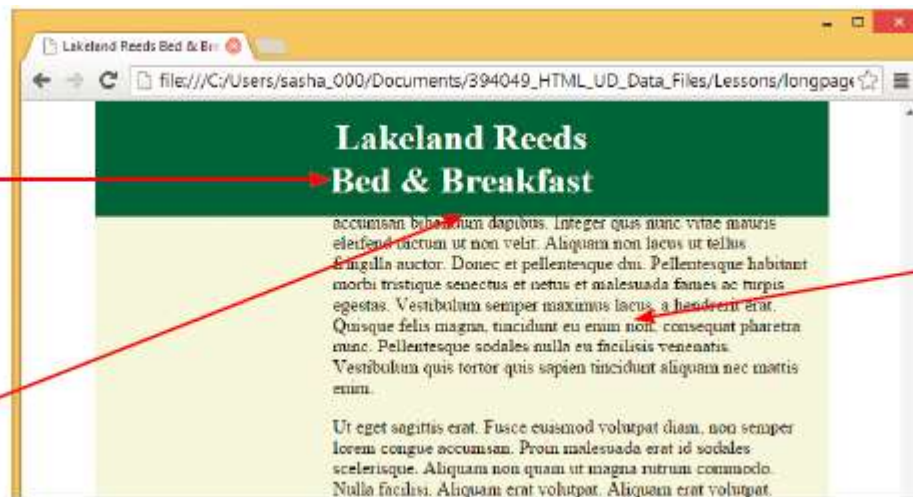
Implement Fixed Positioning (continued)

- Code including `fixed` value and how it is rendered in a browser

```
24  /* headings */
25  header {
26      width: 100%;
27      max-width: 640px;
28      position: fixed;
29      background-color: darkgreen;
30  }
```

header element
remains in the
same place within
the browser window
as you scroll down
the page

article text
moves behind the
header element
as you scroll



Lorem ipsum text
fills the article
element to simulate
a long web page

Unit D

Demo (Fixed Positioning)

css_fixed_positioning.html

A. This div element does not move while the rest of A's scroll down.

A **You can run, but you can't hide.**



A

A

A

A

A

A

A

A.....

A

Stays in place as you scroll down.

Critical Thinking

Discuss fixed positioning with students, especially which elements it might be suitable for. Then have students name some problems that might come up for users when elements are placed using fixed positions.

Implement Relative Positioning

- Relative positioning: adjusting default position of an element
 - Preserves space allotted to element in default page flow
 - Set position property to `relative`
 - Style exact location using the properties
 - `left` and `right` for horizontal positioning
 - `top` and `bottom` for vertical positioning

Unit D

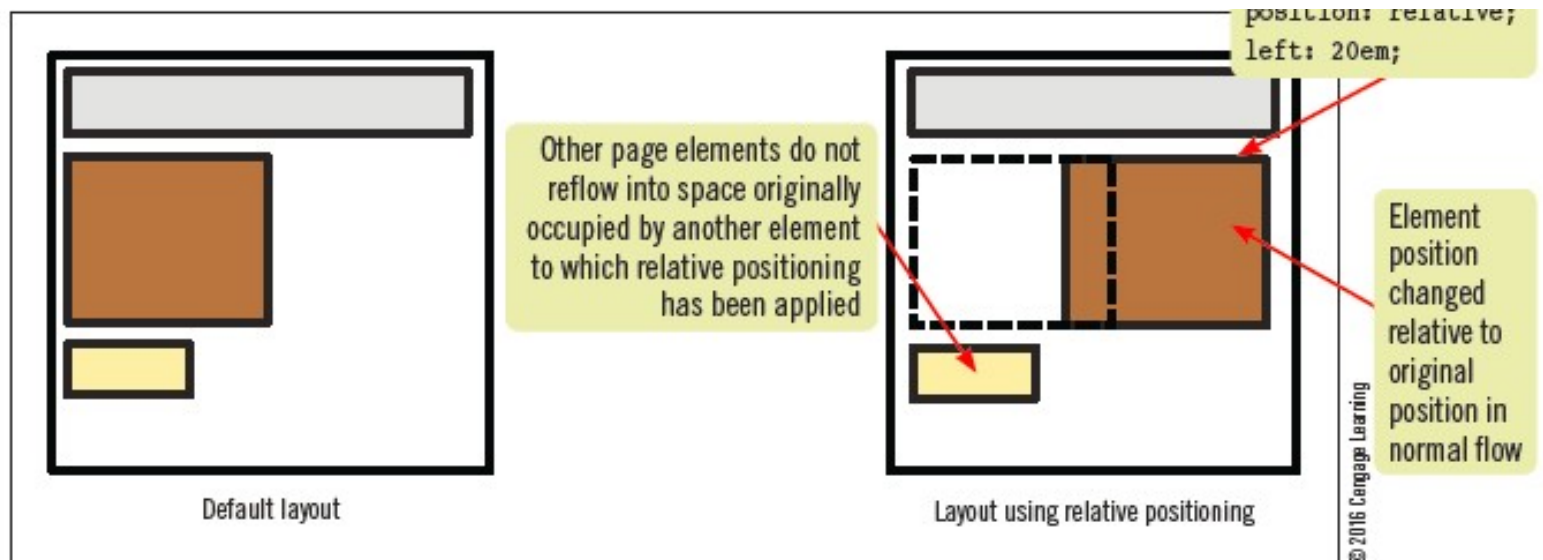
Implement Relative Positioning (continued)

- CSS properties and values for relative positioning

property	value	description
position	relative	Moves element relative to its original position but preserves the space reserved for the element in the normal flow
top, bottom	value in ems, pixels, or another supported unit, or percent of the height of the browser window	Moves the element the specified distance down from the top edge (top) or up from the bottom edge (bottom) of the closest ancestor element that is also positioned; the default value, 0, leaves the element in its original vertical position
left, right	value in ems, pixels, or another supported unit, or percent of the width of the browser window	Moves the element the specified distance from the left edge (left) or from the right edge (right) of the closest ancestor element that is also positioned; the default value, 0, leaves the element in its original horizontal position

Implement Relative Positioning (continued)

- Implementing relative positioning and page flow

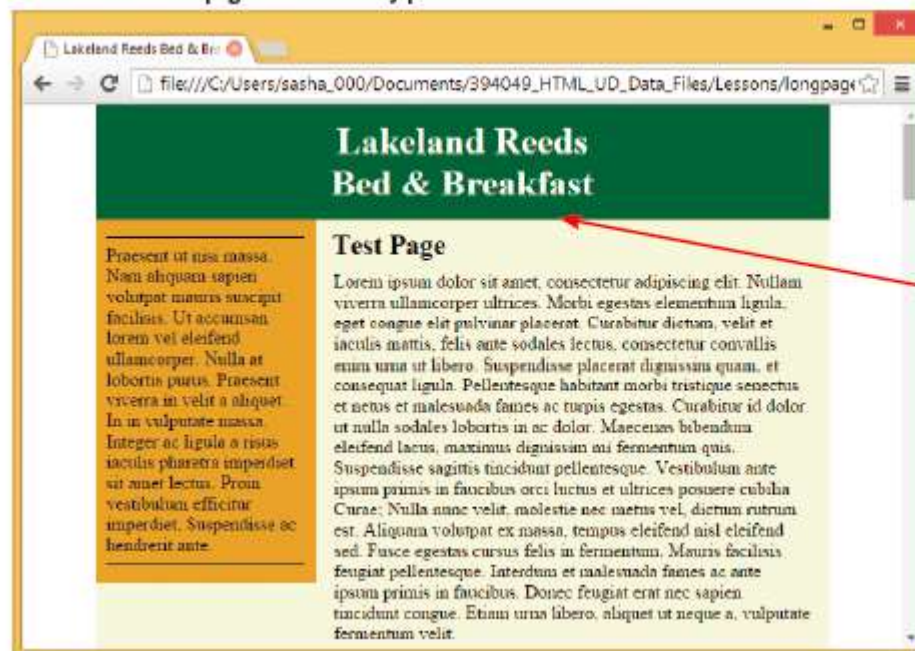


Unit D

Implement Relative Positioning (continued)

- Code using relative positioning and how it is rendered in a browser

```
17  /* body and page container */
18  .container {
19      max-width: 640px;
20      margin: 0 auto;
21      background-color: beige;
22      position: relative;
23      top: 6.2em;
24  }
```



Top of container element is below header element as a result of relative positioning

Unit D

Demo Relative Positioning

css_position_relative.html

These are my relatives, yet they are push relatively to the left.

An element with `position: relative;` is positioned relative to its normal position:

I move to right 100px from the left.

```
<style>
div.myrelatives {
  position: relative;
  left: 100px;
  border: 1px solid red;
}
</style>
```


Unit D

Classroom Activities

Can implementing relative positioning cause elements to overlap? Explain why or why not. How can you deal with such overlap if it occurs?

Yes.

Control stacking order

- Stacking elements: positioning elements so that they can overlap
 - Additional possibilities for layouts
 - Applies only to positioned element
 - Elements placed in new layer
 - Requires careful planning
- Stacking order controlled by values assigned to `z-index` property

Control Stacking Order (continued)

- Stacking positioned elements



Control Stacking Order (continued)

- Code to stack and how it is rendered in a browser

```
26  /* headings */
27  header {
28      width: 100%;
29      max-width: 640px;
30      position: fixed;
31      background-color: darkgreen;
32      z-index: -1;
33  }
```



container element
displayed on top of
header element
because header
has lower z-index
value



Demo (Control Stacking)

css_stacking.html



This is my school.

Because the image has a z-index of -1, it will be placed behind the text.

```
#img1 {  
  position: absolute;  
  left: 10px;  
  top: 0px;  
  z-index: -1;  
}
```

```
#img2 {  
  position: absolute;  
  left: 0px;  
  top: 33px;  
  z-index: -1;  
}
```


Classroom Activities

You are creating a website for a historical society, and they want you to annotate a map of the Shenandoah Valley to indicate the locations of Civil War battles that took place in the valley. The location of each battle should be indicated by an X of a specified color. Can you achieve this without making changes to the original map by using stacked elements? Explain why or why not.

Yes and using overlap attribute



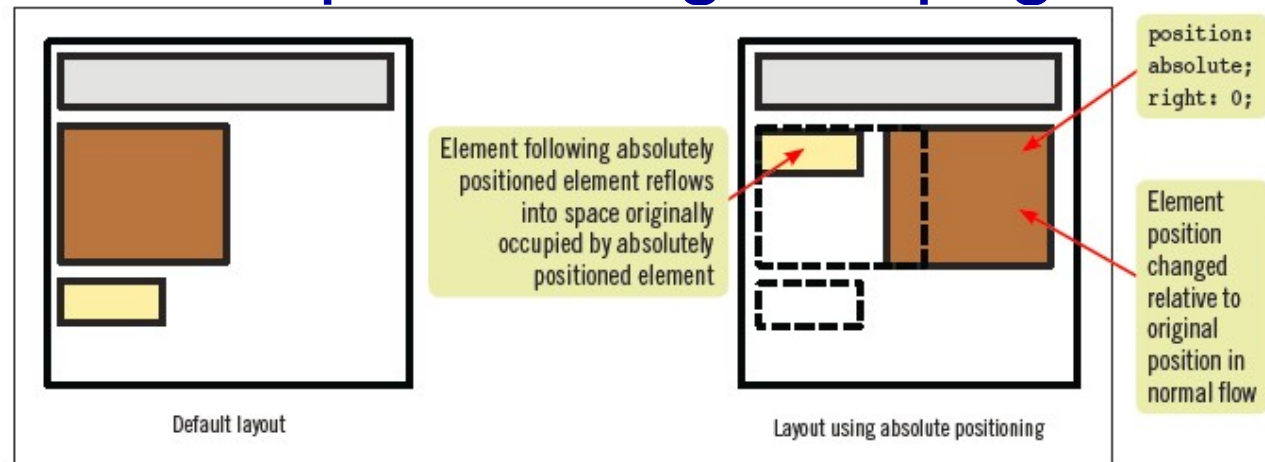
Implement Absolute Positioning

Absolute positioning: takes element out of page flow entirely

- Allows other elements to flow into space element would have occupied
- Set `position` property to `absolute`
- Style exact location using `left`, `right`, `top`, and `bottom` properties
- Location is calculated relative to closest ancestor element that has `position` applied to it

Implement Absolute Positioning (continued)

- Absolute positioning and page flow



- Properties and values: CSS properties and values used in absolute positioning

property	value	description
position	absolute	Removes element from normal flow and positions it relative to the closest ancestor element that is also positioned
top, bottom	value in ems, pixels, or another supported unit, or percent of the height of the browser window	Moves the element the specified distance down from the top edge (top) or up from the bottom edge (bottom) of the closest ancestor element that is also positioned; the default value, 0, leaves the element in its original vertical position
left, right	value in ems, pixels, or another supported unit, or percent of the width of the browser window	Moves the element the specified distance right from the left edge (left) or left from the right edge (right) of the closest ancestor element that is also positioned; the default value, 0, leaves the element in its original horizontal position

Unit D

Implement Absolute Positioning (continued)

- Code to absolutely position an element and the result in a browser

```
39 h2 {  
40   padding: 0.4em 0.6em;  
41 }  
42 p.logo {  
43   padding: 0.2em;  
44   border: 2px solid beige;  
45   color: goldenrod;  
46   position: absolute;  
47   top: 2em;  
48   left: 2em;  
49 }
```

Logo placeholder
text placed next to
main heading using
absolute positioning



Unit D

Demo

(Absolute Positioning)

[css_positioning_absolute.html](#)

My power is absolute without question.

Absolute Example.

This div element has
position: absolute;

```
div.absolute {  
  position: absolute;  
  top: 20%; /*distance from the top edge*/  
  right: 50%; /*left from the right edge*/  
  width: 100px;  
  height: 100px;  
  border: 3px solid red;  
}
```

Summary

- Box model: used by CSS to represent characteristics of elements
 - Treats element as rectangular box
 - Defines border, margin and padding
- Margins of adjacent elements can collapse into one margin
- Specify `border width`, `style`, and/or `color` of any or all sides

Summary (continued)

- Use multiple values to assign `margin` and `padding` values to different sides
- Use `float` to change the default page flow
- Use `clear` to create a more precise layout
- Use `float`, `clear` and `width` to create multicolumn layouts

Summary (continued)

- Fixed positioning: keeps an element in the same position, even when the page is scrolled
- Relative positioning: adjusting location of element relative to its page flow location using position property
 - Space preserved

Summary (continued)

- Use `z-index` to create a stacking order for elements
 - Stacked elements are placed in new layers
- Absolute positioning: takes elements out of page flow entirely
 - Other elements reflow into empty space