CSE 154

LECTURE 4: PAGE SECTIONS AND THE CSS BOX MODEL
### The vertical-align property

<table>
<thead>
<tr>
<th>property</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vertical-align</td>
<td>specifies where an inline element should be aligned vertically, with respect to other content on the same line within its block element's box</td>
</tr>
</tbody>
</table>

- can be `top`, `middle`, `bottom`, `baseline` (default), `sub`, `super`, `text-top`, `text-bottom`, or a length value or `%`
  - `baseline` means aligned with bottom of non-hanging letters
Vertical Align

```html
img { vertical-align: bottom }
```

```html
img { vertical-align: middle }
```

```html
img { vertical-align: top }
```
Motivation for page sections

• want to be able to **style** individual elements, groups of elements, sections of **text** or of the page

• (later) want to create complex page layouts
The HTML id attribute

<p>Spatula City! Spatula City!</p>
<p id="mission">Our mission is to provide the most spectacular spatulas and splurge on our specials until our customers <q>explode</q> with splendor!</p>

- allows you to give a unique ID to any element on a page
- each ID must be unique; can only be used once in the page
Linking to sections of a web page

Visit textpad.com to get the TextPad editor.

View our Mission Statement

• a link target can include an ID at the end, preceded by a #

• browser will load that page and scroll to element with given ID
## CSS ID selectors

### Code Example

```css
#mission {
    font-style: italic;
    font-family: "Garamond", "Century Gothic", serif;
}
```

### Output

Spatula City! Spatula City!  
*Our mission is to provide the most spectacular spatulas and splurge on our specials until our customers ”explode” with splendor!*

- applies style only to the paragraph that has the ID of `mission`
- element can be specified explicitly: `p#mission {`
The HTML class attribute

- classes are a way to group some elements and give a style to only that group ("I don't want ALL paragraphs to be yellow, just these three...")

- unlike an id, a class can be reused as much as you like on the page
CSS class selectors

```css
.special { /* any element with class="special" */
  font-weight: bold;
}
p.shout { /* only p elements with class="shout" */
  color: red;
  font-family: cursive;
}
```

Spatula City! Spatula City!
See our spectacular spatula specials!
Today only: satisfaction guaranteed.

- applies rule to any element with class `special`, or a `p` with class `shout`
Multiple classes

Spatula City! Spatula City!

See our spectacular spatula specials!

Satisfaction guaranteed.

We'll beat any advertised price!

- an element can be a member of multiple classes (separated by spaces)
CSS for following examples

```css
.special {
  background-color: yellow;
  font-weight: bold;
}

.shout {
  color: red;
  font-family: cursive;
}
```

- for the next several slides, assume that the above CSS rules are defined
Sections of a page: `<div>`

*a section or division of your HTML page (block)*

```html
<div class="shout">
  <h2>Spatula City! Spatula City!</h2>
  <p class="special">See our spectacular spatula specials!</p>
  <p>We'll beat any advertised price!</p>
</div>
```

Spatula City! Spatula City!

See our spectacular spatula specials!

We'll beat any advertised price!

- a tag used to indicate a logical section or area of a page
- has no appearance by default, but you can apply styles to it
CSS context selectors

```
selector1 selector2 {  
  properties
}
```

- applies the given properties to `selector2` only if it is inside a `selector1` on the page

```
selector1 > selector2 {  
  properties
}
```

- applies the given properties to `selector2` only if it is `directly` inside a `selector1` on the page (`selector2` tag is immediately inside `selector1` with no tags in between)
## Context selector example

<table>
<thead>
<tr>
<th>HTML</th>
<th>CSS</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;p&gt;Shop at &lt;strong&gt;Hardwick's Hardware&lt;/strong&gt;...&lt;/p&gt;</code></td>
<td><code>li strong { text-decoration: underline; }</code></td>
<td>Shop at <strong>Hardwick's Hardware</strong>...&lt;br&gt;&lt;br&gt;• The <strong>best</strong> prices in town!&lt;br&gt;• Act while supplies last!</td>
</tr>
</tbody>
</table>
## More complex example

```html
<div id="ad">
  <p>Shop at <strong>Hardwick's Hardware</strong>...<p>
  <ul>
    <li class="important">The <strong>best</strong> prices!</li>
    <li>Act <strong>while supplies last!</strong></li>
  </ul>
</div>
```

### CSS

```css
#ad li.important strong { text-decoration: underline; }
```

Shop at **Hardwick's Hardware**...

- The **best** prices!
- Act **while supplies last!**
Inline sections: <span>

an inline element used purely as a range for applying styles

Spatula City! Spatula City!
See our spectacular spatula specials!
We'll beat any advertised price!

• has no onscreen appearance, but you can apply a style or ID to it, which will be applied to the text inside the span
The CSS Box Model

• for layout purposes, every element is composed of:
  ◦ the actual element's **content**
  ◦ a **border** around the element
  ◦ **padding** between the content and the border (*inside*)
  ◦ a **margin** between the border and other content (*outside*)

• width = content width + L/R padding + L/R border + L/R margin
• height = content height + T/B padding + T/B border + T/B margin
  ◦ **IE6 doesn't do this right**
Document flow - block and inline elements
CSS properties for borders

```css
h2 {
  border: 5px solid red;
}
```

This is a heading.

<table>
<thead>
<tr>
<th>property</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>border</td>
<td>thickness/style/color of border on all 4 sides</td>
</tr>
</tbody>
</table>

- **thickness** (specified in px, pt, em, or thin, medium, thick)
- **style** (none, hidden, dotted, dashed, double, groove, inset, outset, ridge, solid)
- **color** (specified as seen previously for text and background colors)
More border properties

<table>
<thead>
<tr>
<th>property</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>border-color, border-width, border-style</code></td>
<td>specific properties of border on all 4 sides</td>
</tr>
<tr>
<td><code>border-bottom, border-left, border-right, border-top</code></td>
<td>all properties of border on a particular side</td>
</tr>
<tr>
<td><code>border-bottom-color, border-bottom-style, border-bottom-width, border-left-color, border-left-style, border-left-width, border-right-color, border-right-style, border-right-width, border-top-color, border-top-style, border-top-width</code></td>
<td>properties of border on a particular side</td>
</tr>
</tbody>
</table>

[Complete list of border properties](#)
Border example 2

```css
h2 {
  border-left: thick dotted #CC0088;
  border-bottom-color: rgb(0, 128, 128);
  border-bottom-style: double;
}
```

This is a heading.

- each side's border properties can be set individually
- if you omit some properties, they receive default values (e.g. border-bottom-width above)
Rounded corners with border-radius

```
p {
    border: 3px solid blue;
    border-radius: 12px;
    padding: 0.5em;
}
```

This is a paragraph.

This is another paragraph.
It spans multiple lines.

- each side's border radius can be set individually, separated by spaces
CSS properties for padding

<table>
<thead>
<tr>
<th>property</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>padding</td>
<td>padding on all 4 sides</td>
</tr>
<tr>
<td>padding-bottom</td>
<td>padding on bottom side only</td>
</tr>
<tr>
<td>padding-left</td>
<td>padding on left side only</td>
</tr>
<tr>
<td>padding-right</td>
<td>padding on right side only</td>
</tr>
<tr>
<td>padding-top</td>
<td>padding on top side only</td>
</tr>
</tbody>
</table>

Complete list of padding properties
# CSS properties for margins

<table>
<thead>
<tr>
<th>property</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>margin</td>
<td>margin on all 4 sides</td>
</tr>
<tr>
<td>margin-bottom</td>
<td>margin on bottom side only</td>
</tr>
<tr>
<td>margin-left</td>
<td>margin on left side only</td>
</tr>
<tr>
<td>margin-right</td>
<td>margin on right side only</td>
</tr>
<tr>
<td>margin-top</td>
<td>margin on top side only</td>
</tr>
</tbody>
</table>

Complete list of margin properties
CSS properties for dimensions

This paragraph uses the first style above

An h2 heading

<table>
<thead>
<tr>
<th>property</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>width, height</td>
<td>how wide or tall to make this element (block elements only)</td>
</tr>
<tr>
<td>max-width, max-height, min-width, min-height</td>
<td>max/min size of this element in given dimension</td>
</tr>
</tbody>
</table>
Centering a block element: auto margins

```css
p {
  margin-left: auto;
  margin-right: auto;
  width: 750px;
}
```

- to center inline elements within a block element, use `text-align: center;`
- works best if `width` is set (otherwise, may occupy entire width of page)