Layout in CSS

configuring and positioning html elements
Goals

• To learn how to position HTML elements on a webpage using CSS
Configuring HTML elements

– width property
  • Configures the width of an element’s content

    h1 { width: 80%; }  

– min-width property
  • Configures minimum width of an element

– max-width property
  • Configures the maximum width of an element

– height property
  • Configures the height of an element

    h1 { height: 100px; }
The Box Model

• Each HTML element in a document is considered a rectangular box:
  
  • Content
    – Text & web page elements in the container
  
  • Padding
    – Area between the content and the border
  
  • Border
    – Between the padding and the margin
  
  • Margin
    – Determines the empty space between the element and adjacent elements
The Margin Property

• Configure the space between elements

Syntax examples:

• configure all margins: `{ margin: 0px }`

• individual margins: `{ margin-left: 20px, margin-right: 20%, margin-top: 0, margin-bottom: 0 }`

• top=bottom, right=left:
  `{margin: 0px 20px }`
Other properties

• padding: area between border and content works similar to margin
• border: in addition to size, can also set the border style (e.g. dotted, dashed, solid) and the width

Examples:

h1 {border: 20px dotted }
p { border-style: dotted}
h2 {border-left-width: 20px}
Normal and nested flow

• Browser display of elements in the order they are coded in the Web page document
Examples

• Normal flow:
  <div>
    <p>part of webpage</p>
  </div>
  <div>..</div>

• Nested flow:
  <div>
    <div>
      <div>
        <p>This box is contained in another box</p>
      </div>
    </div>
  </div>
Block vs inline elements

• HTML elements can be grouped into block elements and inline elements
• Block elements always appear on a new line and include e.g.: `<p>`, `<h1>`, `<ul>`, `<div>`
• Inline elements appear without starting a new line: `<strong>`, `<b>`, `<em>`, `<span>`, `<img>`

(whether an element has the "block" or "inline" property can be changed in CSS using the `display` property)!
The float property

- Normally block elements flow in a column
- Float can be used to place block elements next to each other e.g.:
  - text next to an image
  - navigation bar next to the content pane
- Needs to be used in combination with the width property so that floats fit next to one another
Specifics on floating elements

• A left-floated box will shift to the left until its leftmost margin edge (or border edge if margins are absent) touches either the edge of the containing block, or the edge of another floated box
• If the size of the floated box exceeds the available horizontal space, the floated box will be shifted down
• Non-positioned, non-floated, block-level elements act as if the floated element is not there, since the floated element is out of flow in relation to other block elements
• The root element (<html>) cannot be floated
• An inline element that is floated is converted to a block-level element
Example

```html
<img src="slide0.jpg" style="float:left"/>
<div style="100px"
style="float:right">
  <p>Some text</p>
</div>
```
Simple layout example

```css
#wrap {
    width: 750px;
    margin: 0 auto;
    background: #99c;
}

nav {
    margin: 0;
    padding: 10px;
    background: #c99;
}

article {
    float: left;
    width: 480px;
    padding: 10px;
    background: #9c9;
}

aside {
    float: right;
    width: 230px;
    padding: 10px;
    background: #0c9;
}

<div id="wrap">
    <nav>...</nav>
    <article>...</article>
    <aside>...</aside>
</div>
```
Layout using absolute positioning

• By default, elements are positioned in normal flow
• Can be changed using the position property:

<table>
<thead>
<tr>
<th>Value</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>static</td>
<td>Default value; the element is rendered in normal flow</td>
</tr>
<tr>
<td>fixed</td>
<td>Configures the location of an element within the browser viewport; the element does not move when the page is scrolled</td>
</tr>
<tr>
<td>relative</td>
<td>Configures the location of an element relative to where it would otherwise render in normal flow</td>
</tr>
<tr>
<td>absolute</td>
<td>Configures the location of an element outside of normal flow precisely in its container element</td>
</tr>
</tbody>
</table>

Table 7.3 The position Property
Relative positioning

• Changes the location of an element in relation to where it would otherwise appear in normal flow

```css
p {
    position: relative;
    left: 30px;
    font-family: Arial, sans-serif;
}
```
Absolute positioning

• Precisely specifies the location of an element outside of normal flow in relation to its first parent non-static element

```css
p { position: absolute;
  left: 200px;
  top: 100px;
  font-family: Arial, sans-serif;
  width: 300px; }
```
Example

```css
#wrap{
  position: relative;
  width: 800px;
  margin: 0px auto;
  background-color: #00cc00;
}
nav {
  position: absolute;
  left: 0;
  margin: 0px;
  padding: 5px;
  width: 200px;
  background-color: #00ccff;
}
article {
  position: absolute;
  left: 220px;
  width: 570px;
  margin: 0px;
  padding: 5px;
  background-color: #aaccff;
}
```

```html
<div id="wrap">
  <nav>..</nav>
  <article>..</article>
</div>
```
Vertical Navigation with an Unordered List

```html
<nav>
  <ul>
    <li><a href="index.html">Home</a></li>
    <li><a href="menu.html">Menu</a></li>
    <li><a href="directions.html">Directions</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</nav>
```

- CSS removes the list marker and underline:

```css
nav ul { list-style-type: none; }
nav a { text-decoration: none; }
```
display Property

• Configures how and if an element is displayed
  – display: none;
    • The element will not be displayed.
  – display: block;
    • The element is rendered as a block element – even if it is actually an inline element, such as a hyperlink.
  – display: inline;
    • The element will be rendered as an inline element – even if it is actually a block element – such as a <li>.
Horizontal Navigation with an Unordered List

```html
<nav>
  <ul>
    <li><a href="index.html">Home</a></li>
    <li><a href="menu.html">Menu</a></li>
    <li><a href="directions.html">Directions</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</nav>
```

- CSS removes the list marker, removes the underline, adds padding, and configures the list items for inline display.

  ```css
  nav ul { list-style-type: none; }
  nav a { text-decoration: none;
          padding-right: 10px; }
  nav li { display: inline; }
  ```
Additional resources

• Web design book: Chapter 7