

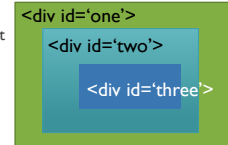
COMP 20 WEB PROGRAMMING

WEEK 4
LISA DIORIO
LISA.DIORIO@TUFTS.EDU

1

POSITION

- Absolute
 - Relative to the first parent element with *position* set
- Fixed
 - Relative to the page
 - May also want to use "z-index" property
- Relative
 - Where it would normally go on the page
 - Need to set this so that a parent element is established



2

MARGIN AND PADDING

- top, left, bottom, right
 - padding: 2px 3px 4px 5px;
- Set with two values to repeat for bottom and right
 - padding: 2px 3px;
- Set with one value to repeat for all
 - padding: 2px;
- Tricks

■ Set left and right to 0	centers a block
■ Set top and bottom to 0	full column height
■ auto will always auto calculate	
■ margin: 0 auto	common technique to center a block

3

MORE BOX PROPERTIES

- float (left, right)
- display (inline, block, inline-block, none)
- height, width
- overflow (overflow-x, overflow-y)
- visibility
- z-index

4

TRANSITIONS

- New to CSS3
- transition-property
- transition-duration
- transition-delay
- transition-timing-function

5

WEBKIT

- Special extensions
- Prefixed with: -webkit-
- Some properties require webkit and non-webkit versions to work on all browsers

6

WHAT IS RESPONSIVE DESIGN?

- Responsive sites have all information accessible and easy to read, regardless of the browser width i.e., the site is responsive to the browser size.
- It should not be an afterthought
 - Responsive first design
 - Better, Mobile first design

7

7

RESPONSIVE DESIGN: WHY IS IT IMPORTANT



8

MOBILE USAGE- SOME STATISTICS

- Consumers now spend more than five hours a day on their smartphones, and a recent survey found 88% growth year over year in time spent watching videos on a smartphone (AdWeek)
- Up to 70% of web traffic happens on a mobile device (CIO Dive, 2018)
- 57% of all U.S. online traffic now comes from smartphones and tablets (BrightEdge, 2017)
- 75% of email is opened on a smartphone
- In 2017, 95.1 percent of active Facebook user accounts accessed the social network via a mobile device (Statista, 2018)
- 69% of smartphone users also say that they are more likely to buy from companies with mobile sites that easily address their questions or concerns (BrightEdge, 2017)

9

RESPONSIVE DESIGN: HOW TO

- Viewport
- Adaptive elements
- Media Queries
- Flex boxes
- Mobiletest.me

10

VIEWPORT

- Defines the area of the screen where the browser can render content
- `<meta name='viewport' content='width=device-width, initial-scale=1'>`
- Reflows content to match the device size
- Put this line in the `<head>` section

11

ADAPTING YOUR DESIGN

- Create elements at full width to size with the screen
 - `width=100%`
 - `max-width=100%` or `max-width=value`
 - Example:
 - `img {width: 100%; max-width: 400px;}`
- Create columns or blocks with a `max-width` to enable full width at small device sizes or multi-column at wider device sizes

12

GRID SYSTEMS

- Set up a grid layout and all elements must be a multiple of the grid width
- Adjust the grid sizes at various breakpoints if needed, and the size will automatically resize



13

FLEX BOX

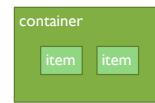
- Flex display controls how items fit and are aligned within a container.

Container properties

`display: flex;`
`flex-direction` – rows or columns
`flex-wrap` – should it wrap
`justify-content` – distributes extra leftover space
`align-items` – can align vertically within a container
`align-content` – aligns rows against each other

Item properties

`order` – order items should appear (if not the order they appear)
`flex-basis` – set size of an item
`flex-grow`, `flex-shrink` – allows some items to be larger or smaller than others
`align-self` – can override “align-items” for one item



14

14

MAKING A SITE RESPONSIVE USING CSS

- `@media` queries override style rules based on browser width
- Syntax:
 - `@media (max-width: 900px) { /* styles go here */ }`
 - `<link rel="stylesheet" href="handheld.css" media="only screen and (max-device-width: 700px)"/>`
- You can specify that styles are only for printed documents, screen readers, screens, etc (all, print, screen, or speech).
- `@media only screen and (max-width: 900px) { /* styles go here */ }`
- Example:

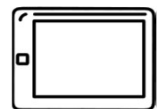

```
@media screen and (max-width: 600px) {
  .no-mobile { display: none; }
}
```
- Can also specify other elements such as orientation (portrait or landscape)
- Determine breakpoints

15

15

MAKING A SITE RESPONSIVE: USEFUL STYLES

- font-size
 - em vs px



	body { font-size: 100%; }	body { font-size: 100%; }
font-size: 1em	The quick brown fox	The quick brown
font-size: 1.2em	The quick brown fox	The quick brown fox
font-size: 1.4em	The quick brown fox	The quick brown fox
font-size: 1.6em	The quick brown fox	The quick brown

© Ryan S. Davis

- Don't forget tablet sizes!

- Mobile simulator:
 - <http://mobiletest.me/>

16

16

MORE STYLES

- width, height
- box-sizing: border-box
- min/max height and width
- float, text-align
- padding, margin
- display



Goal: set up site such that you do not need to make changes via media queries

Ex:
Set width to 100%
Set max-width to 400px

Photos by Unknown Author is licensed under CC BY-SA

17

17

MAKING A SITE RESPONSIVE: DUPLICATE CONTENT

- Sometimes the changes to a section from "desktop" to "mobile" are too drastic
- It doesn't always make sense just to "squeeze" the page and see where things fall
- Example- menus, headers, footers
- One option is to have two div's with versions for different sizes.

```
<div class="no-mobile">Full width stuff goes here</div>
<div class="is-mobile">Mobile friendly stuff goes here</div>
```

```
.is-mobile {display:none;}
@media (max-width:400px) {
  .is-mobile {display:block;
  .no-mobile {display:none;} }
```

18

18

MOBILE MENUS

- Wide text-based menus will simply not be visible or will wrap
- Best to use a "hamburger" menu
- Think about ...
 - Which menu items should be in mobile
 - What browser width to switch to mobile
 - Logo size / menu position
 - Different header
 - Buttons big enough for fingers



19

19

HTML TABLES

- Main tag is <table>
- Tables are made of rows: <tr>
- Rows are made of elements (table data) <td>
- "Header" rows are bold and centered <th>
- Rows are auto sized to the tallest item in the row
- Columns are auto sized to the widest element in the column
- Use for an empty cell

Header	Header	Header	Header

Photos by Unknown Author is licensed under CC BY-SA

20

HTML FORMS

- Main tag is `<form>`
- `<input>` for text, radio, checkbox, submit, reset
- `<select>` for drop-down/list
- `<textarea>` for multi-line text
- Use CSS for positioning
- Use id, name to identify elements in script

Pet Information Form

Name:

Create a password:

Where did you hear about us?

☐ Friend
☐ Internet
☐ Other

Type of pet: ☒ Dog ☐ Cat ☐ Hamster

Alternate type of pet:

Describe the problem:

21

HTML 5

Supersedes HTML4.01

Created in cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).

Cheat sheet at:
<https://tuftsdev.github.io/WebProgramming/notes/html5-cheat-sheet.pdf>

- Originally for "catalog" type markup
- Enhanced mobile development
- Can be reached by users with or without the internet – important for speed for things like online chat
- Can save important data to databases locally
- Graphic and visual effects (formerly required Flash)
- Integrates seamlessly with CSS3
- Accessibility
- Video and audio support built in
- Simpler DocType declaration
- Tags are more descriptive

22

DOCTYPE DECLARATION

- Tells the browser what HTML version the page is using
- Allows for more uniform behavior amongst browsers
- HTML 4.01 DOCTYPE
 - `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">`
- HTML5 DOCTYPE
 - `<!DOCTYPE html>`

23

23

STRUCTURE OF AN HTML5 FILE



24

24

EXAMPLE HTML5 DOCUMENT

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset = "utf-8">
    <title> My HTML5 Page</title>
  </head>
  <body>
    <header>
      <h1>Title for My Page</h1>
    </header>
    <nav>
      <ul>
        <li><a href="#">Home</a></li>
        <li><a href="#">Contact</a></li>
      </ul>
    </nav>
    <article>
      <section>
        <p> Sections can be within an article</p>
      </section>
      <aside>
        <p>An aside may be styled as a sidebar</p>
      </aside>
      <footer> footer stuff here ... </footer>
    </article>
  </body>
</html>

```

25

DEPRECATED TAGS (DON'T USE THEM)

- font, center, b, i, u
- frame, frameset
- applet
- marquee, bgsound
- noscript
- See: <http://www.html-5.com/changes/deprecated/>

26

HTML5 SECTION VS ARTICLE

- From the HTML5 Spec ...
 - **The section element** represents a generic section of a document or application. A section, in this context, is a thematic grouping of content, typically with a heading
 - **The article element** represents a self-contained composition in a document, page, application, or site and that is, in principle, independently distributable or reusable, e.g. in syndication. This could be a forum post, a magazine or newspaper article, a blog entry, a user-submitted comment, an interactive widget or gadget, or any other independent item of content.

27

HTML 5: NEW DOCUMENT TAGS

- | | |
|---|---|
| ■ Aside | ■ Nav |
| ■ Somewhat related to the rest of the page. | ■ Navigation of the document |
| ■ Header | ■ Dialog |
| ■ header of the document | ■ mark up a conversation. |
| ■ Footer | ■ Figure |
| ■ footer for a document | ■ associate a caption with embedded content |

28

NEW HTML5 FORM ELEMENTS

- Date and time (datetime, date, month, week, time)
- number
- range
- Email
- url
- output

29

29

TRY IT: "CREATE" THIS DOCUMENT IN HTML5



30

30

XML IN BRIEF (WE COULD TAKE A LOT LONGER!)

- Extensible markup language
- For description/storage of data

31

31

XML TECHNOLOGIES

- DTD, Schema
 - describe data
- XSLT
 - transform XML to XML
 - can be used as a stylesheet
- DOM
 - tree-structured view
 - random access to data
- SAX (Simple API for XML)
 - event based API for accessing XML data
 - single pass through data
- Namespaces
 - Help distinguish between data vocabularies

32

32

XML PARTS

- Names
 - start with letter, underscore, colon
 - then letter, underscore, colon, digits, hyphens, periods
 - Cannot start with "xml"
- Whitespace is preserved (unlike HTML)
- Elements: Described by tags
- Element content
 - Simple: text
 - Element: an element contains other elements
 - Mixed: text and other elements
- Attributes
 - Name/value pairs
- Case sensitive

33

33

WELL FORMED XML

- Must have **at least one element**
- Must have only **one root node**
- Must have **end tags**
 - empty elements can use the /> notation
 - i.e., <middle />
- Must be **nested properly**
- Also –
 - Every attribute must have a name and a value and values must be quoted

34

34

EXAMPLE OF XML FILE

```
<?xml version="1.0" encoding="UTF-8"?>
<Listings>
  <Listing type="residential">
    <StreetAddress>
      <StreetNumber>100</StreetNumber>
      <StreetName>Maple Street</StreetName>
    </StreetAddress>
    <City>Rochester</City>
    <State>New York</State>
    <Zip>14623</Zip>
  </Listing>
</Listings>
```

35

35

NOT WELL-FORMED- WHAT IS WRONG?

```
<?xml version="1.0" encoding="UTF-8"?>
<HomeListings>
  <Listing daysListed="30">
    <StreetAddress>
      <StreetNumber>100
      <StreetName>Maple
      <StreetNumber></StreetAddress>
    <City>Rochester</City><State>New
    York</State>
    <Zip>14623</Zip>
    <zoning type="residential">
  </Listing>
</HomeListings>

<CommercialListings>
  <Listing daysListed=73>
    <StreetAddress zoning=business>
      <StreetNumber>150</StreetNumber>
      <StreetName>Main Street</StreetName>
    </StreetAddress>
    <City>Rochester</City><State>New
    York</State>
    <Zip>14623</Zip>
    <zoning type="business">
  </Listing>
</CommercialListings>
```

36

36

VALID XML

- Well-formed XML is syntactically correct
- Valid XML – three criteria
 1. document is well-formed
 2. document schema definition can be located
 3. The structure of the XML data conforms to the specified schema
- The browser will generally report when an XML file is
 - Not well-formed
 - Not valid

37

37

DESCRIBING XML DATA: SCHEMA

- DTD
 - Document Type Definition
 - describes type, sequence and requirements of the data (meta data)
 - can be internal or external
 - `<!DOCTYPE book SYSTEM "...lbooks.dtd">`
- Schema
 - .xsd
 - Formatted as XML
 - Supports namespaces and user defined data types
 - Built-in support for modularity

38

38

STYLING THE DISPLAY OF THE XML FILE

- An XML file will display in most browsers as a hierarchical tree view
- You can use CSS or XSLT to "style" the display
- CSS only allows basic styling
 - Link the stylesheet using a processing instruction inside the XML file
- XSLT goes much further allowing filtering and broader transformations in addition to styling

39

39

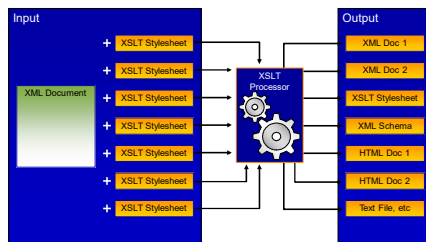
XSLT – EXTENSIBLE STYLESHEET LANGUAGE

- More options than CSS
- Many transformation options
 - XML to XML
 - Can combine with CSS for formatted output
 - XML to HTML

40

40

XSLT TRANSFORMATIONS



41

41

POP QUIZ

1. True or false: XML is the technology of choice when your primary goal is display of information.
2. Which of the following is not a technology associated with XML?
XSLT - ASP - DTD - SAX - CSS
3. True or false: an XML document is considered to be well formed if it has been validated against a DTD.
4. True or false: A document can be well-formed and not valid.
5. True or false: DTD is in XML format but Schema is not
6. True or false: A document can be validated against a DTD or a Schema

7. Consider the following XML code:

1. <Employee>
2. <Name>
3. <FirstName>Bill</FirstName>
4. <LastName>Blass</LastName>
5. </Name>
6. <Title>Manager</Title>
7. </Employee>

Identify the line number that contains each of the following elements:

- | | |
|-------------------|------------------|
| a. Root element | b. Child element |
| c. Parent element | d. End tag |

42

42