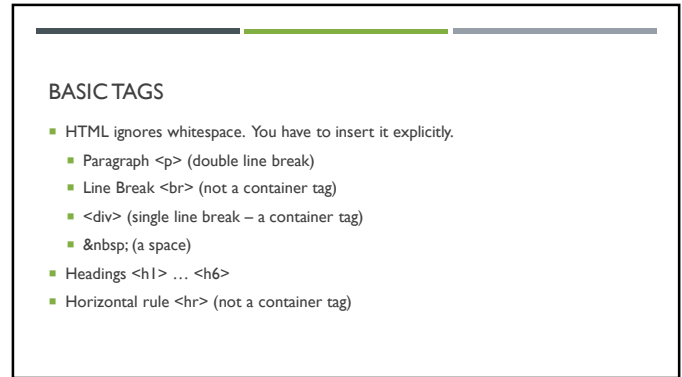
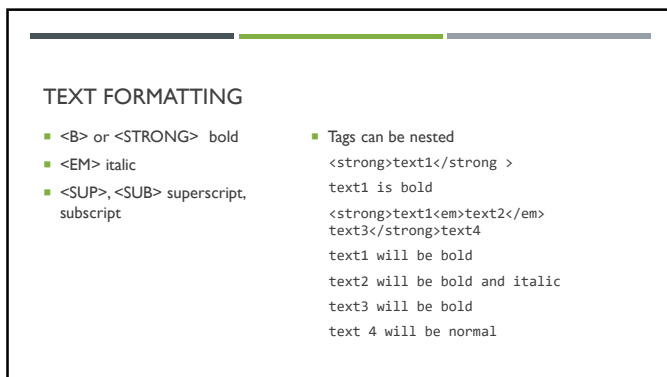




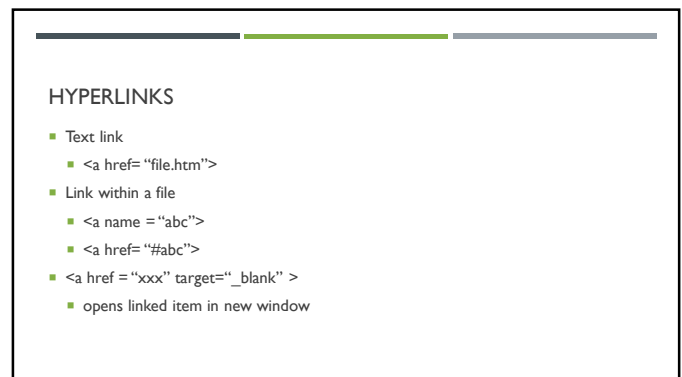
41



42



43



44

EMAIL AND FILE LINKS

- Email link
 - ``
- Link to a file – set href to the file name
 - can be used for downloads
 - file must be uploaded to the server
- Where to find files:
 - If a file in same folder as web page: file.xxx
 - If file is in child folder of web page: folder_name/file.xxx



Photo by Unknown Author is licensed under CC BY 3.0

45

IMAGES

- ``
- img is not a container tag
- Can auto size – may deter performance
- Attributes
 - alt, title
 - height, width
 - align
- Formats: jpg vs png vs WebP
- Finding images
 - unsplash.com
 - Stock photos
 - Creative commons
 - Attribution



Photo by Unknown Author is licensed under CC BY 3.0

46

LISTS

- Unordered (bulleted) ``
- Ordered (numbered) ``
- List item ``
- You can create an indent level by creating another list inside of an ``

47

META TAGS

- Metadata is "data about the data"
- `<meta ... />`
 - Name
 - Description Keywords
 - Author Viewport
 - Content
- Redirect
 - `<meta http-equiv="refresh" content="5;URL=https://www.newsite.com/" />`

48

CSS (CASCADING STYLE SHEETS)

- Define style instructions for a page
- Developed by the W3C
- Allows formatting to be separate from content
- A style sheet applies only to the current page
 - However, a style sheet file can be shared by all pages in a site.
- CSS is a series of formatting rules for your html elements

49

CSS RULES

- Each rule has a selector and a set of property/value pairs
 selector{style-property1:value;style-property2:value}
- The **selector** indicates what the rule will apply to.
- The **properties** are the particular style characteristics that are being modified.
- The **value** is the new value for that property.

50

EXAMPLE

```
h1 {text-align:center; color: #ff0000;}
```

- Selector: h1
- Property: text-align, Value: center
- Property: color, Value: #ff0000
- This rule states that all h1 tags should be centered on the page and colored red.
- “!important” at the end of any rule indicates priority
 - text-align:center !important;

51

COLORS

- Color names: 'blue', 'red', etc. should NO LONGER be used
- Hexadecimal colors for HTML or CSS: RGB, red green blue
 - Begin with # and enclose in quotes
 - Each section ranges from 00 to FF
- Example, red is: #FF0000
- Or, rgb() color
 - rgb(12, 120, 255)
- Or, rgba() color to add transparency
 - rgba(12, 120, 255, .4)

52

WHERE TO SET COLORS IN CSS

- color (text color)
- background-color
- Border colors
 - border: 2px solid #123456;

53

WHERE DO STYLES LIVE?

- External stylesheet `<link rel='stylesheet' type = "text/css" href="file.css">`
 - File contains rules only
 - Use link tag to attach to a file
 - Can be shared by several pages
- Internal stylesheet
 - Use `<style>` tag
 - Takes precedence over external stylesheet
- Inline
 - "at the tag"
 - `<p style="color:#ff0000">`
- The closer a rule is to the selected element, the stronger the precedence

54

CSS SELECTORS

- Selectors can be:
 - An HTML tag
 - A style class – starts with a period (.)
 - An id – starts with a #
 - A pseudo-element – starts with a colon (:before)

55

SELECTOR EXAMPLES

- tag rule applies to the specified tag
- tag1,tag2 rule applies to all of the specified tags (no space after comma)
- tag1 tag2 rule applies when both tags are set
ie, "h1 b" applies when "`<h1> Text is h1 and bold</h1>`"
- tag [attr=value] rule applies to specified tag only when the attribute is set to the given value `<tag attr="value" ...`
- .x rule applies when class="x" `<tag class="x" ...`
- tag1.x rule applies for all tag1 when its class="x" `<tag1 class="x" ...`
- a.x:hover rule applies to links that specify class "x" `<a class="x" href="..."`
- div#x rule applies when id is x `<div id="x" ...`

56

HTML BLOCKS AND CSS

- Blocks are often used to facilitate styling
 - <p> Double line break
 - <div> Single line break
 - No line break
- Add inline styling or add a class or id
- Other than line spacing, no other default styling exists for these elements.



Photo by Unknown Author is licensed under CC BY

57

METRICS FOR SIZES

- px, pt, pc
- in, cm, mm
- %
- auto
- inherit
- em
 - relative to the current font size (2em => 2 times the size of the current font)

height, max-height, min-height
width, max-width, min-width

Must set display to block or inline-block

58

FONTS & TEXT

- color
- text-align
- letter-spacing
- line-height
- text-transform (case)
- text-decoration
- text-shadow
- font-family
- font-size
- font-style (italic)

59

STYLESHEET EXERCISE

- What does each line do?
- Build an HTML doc to meet each of the styles

```
<style type = "text/css" >
  h1 {color : #234567;
      font-family:georgia,times}
  p {font-size:16px; line-height: 1.4;}
  p.center {font-size:16pt;
            text-align:center}
  .quoted {font-style: italic;
            font-size:14pt;
            padding-left: 30px;
            padding-right: 30px;
            box-sizing: border-box;}
  a {font-family:times; font-size: 2em;
      color:#ff0000; text-decoration:none}
  a:hover {color : #00f844;}
</style>
```

60

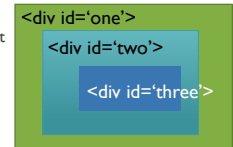
BACK TO CSS ... THE BOX MODEL

- Border - around entire item
- Padding - space between content and boundary of item
- Margin - space between items
 - Trick: set margin left and right both to "auto" to center text.
 - margin: 3px auto ;
- box-sizing: border-box

61

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



62

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

63

MORE BOX PROPERTIES

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

64

TRANSITIONS

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

65

WEBKIT

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

66

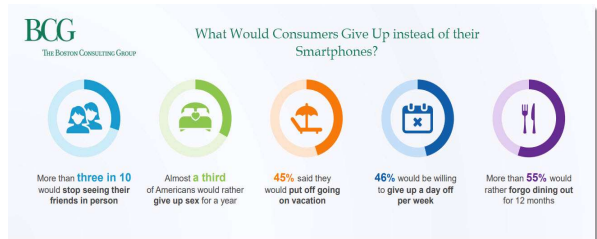
WHAT IS RESPONSIVE DESIGN?

- 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

67

67

RESPONSIVE DESIGN: WHY IS IT IMPORTANT



68

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 (CIODive, 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)

69

69

RESPONSIVE DESIGN: HOW TO

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

70

70

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

71

71

ADAPTING YOUR DESIGN

- Create elements at full width to size with the screen
 - `width=100%`
 - `max-width=100%` or `max-width=value`
 - Example:
 - `img {max-width: 100%; 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

72

72

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 and the size will automatically resize beautifully



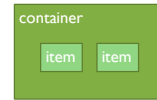
73

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

74

74

MAKING A SITE RESPONSIVE: 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:

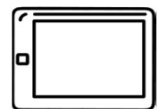

```
@media screen and (max-width: 600px) {
  .no-mobile { display: none; }
}
```
- Can also specify other elements such as orientation (portrait or landscape)
- Determine breakpoints

75

75

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

• Don't forget tablet sizes!

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

76

76

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

77

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;}
```

78

MOBILE MENUS

- Wide text-based menus will simply not be visible
- Best to use a "hamburger" menu
- Think about ...
 - Which menu items should be in mobile
 - What browser width to switch to mobile
 - Logo size
 - Different header
 - Buttons big enough for fingers
- Use a plugin or theme support or switch in an alternate div on mobile sizes



79