Design Document: CSS Beyond Basics

Class Description

Learn more advanced techniques for styling your web page using the programming language CSS by exploring the box model, margins, padding, positioning and other layout concepts.

**Prerequisites:** Capable of using a web browser (i.e. Google Chrome, Internet Explorer, Safari or Firefox).

**Curriculum Track**

Software & Apps

**Audience**

Adults

**Course Length**

90 minutes

**Training Method**

Instructor-led hands-on

**Purpose**

This program will introduce the concept of the box model and how to apply it to an HTML document followed by several techniques for styling each bit of content using images, fonts, and positioning.

Equipment Requirements

Projector and projection screen; computers with internet access for the instructor and each participant; laser pointer (recommended)

Software Requirements

One of the following. Windows 7, Linux, OS X, a text editor or an in-browser code editor like Codepen.io

Material Requirements

Pens or pencils, activity sheets, handouts, participant surveys

**Learning Objectives**

At the end of the session, learners will:

* Learn how to apply Semantic Elements
* Apply font styles
* Understand the Box Model
* Position elements on the page

**Assessment Technique(s)**

Successful completion of class activities

Content Outline

**Agenda (3 mins)**

* Outline the following topics that will be covered in class:
  + Terminology & Descriptions
  + Page Structure: HTML5 Semantic Elements
  + Web Typography
  + Box Model
  + Positioning Elements

**Topics, Talking Points, and Activities (85 mins.)**

* Terminology & Descriptions
  + Define the following terms for participants:
    - **Semantic Elements:** HTML elements with specific meaning for defining the content
    - **Header:** an element specifying a header for a document or section
    - **Nav:** an element defining a set of navigational links
    - **Section:** an element defining a section in a document
    - **Article:** an element for specifying independent or self-contained content
    - **Aside:** an element defining content aside from the surrounding content (like a sidebar)
    - **Footer:** an element for specifying the footer of a document or section
    - **Box Model:** defines the elements as a rectangular box with height, padding, borders and margins
    - **Margin:** sits outside the edge of a border
    - **Padding:** the space between the border of a box and its content
    - **Border:** separates the edge of one box from another
    - **Web Font:** a file of a typeface hosted on a server

**ACTIVITY**: Have participants complete **Activity #1** on the *Activity Sheet*

* Page Structure: HTML5 Semantic Elements
  + Semantic elements define different parts of a web page
  + Before the release of HTML5, developers used their own id/class name to style elements
  + The development of semantic elements has led to a sharable/reusable language
  + Semantic elements also allow search engines to more easily identify the page content
  + There are more than a dozen sematic elements available in HTML5
  + The most commonly used are <header>, <footer>, <nav>, <section>, <article> and <aside>
  + Live demo how to add semantic elements to an HTML document
  1. Confirm the students have their Codepen account open
  2. Open <http://codepen.io/Meggen/pen/EWxbvQ?editors=1000#0>
  3. Demonstrate how different semantic elements define pieces of the page content

**ACTIVITY**: Have participants complete **Activity #2** on the *Activity Sheet*

* Typography
* Define the following
  1. Typeface
     1. Typefaces describe the overall look of the characters contained within the font
  2. Font
     1. A font is a file that contains a typeface
* Describe what a Web Font is
  + - Web fonts allow designers/developers to use fonts not installed on the visitors’ computer
    - To used them you must include or reference the font hosted on a web server
    - The font is automatically downloaded from the server and used as your display text
    - If the font is hosted on your server the CSS3 rule *@font-face* points to that font file
    - Google and Typekit offer font libraries that can be imported into your CSS document
    - **Example:** @import url(‘https://fonts.googleapis.com/css?family=Raleway’);
    - To apply the font to your site you must use the *font-family* property
    - **Example:** body {font-family: ‘Raleway’, sans-serif;}*(Sans-serif is an alternative font if the server hosting Raleway is not available. If your first choice is not available, your text will be displayed using the default sans-serif available within the computer.)*
* Demonstrate how to set the typeface

1. Open <http://codepen.io/Meggen/pen/vxELqy?editors=1100>
2. Demonstrate how to import a font from Google
3. Enter <https://fonts.google.com/> in the address bar of the browser
4. Search for the typeface Raleway
5. With the new display window open, click **SELECT THIS FONT** in the upper right corner
6. Click the black bar at the bottom titled **Family Selected**
7. Click **@IMPORT** for the import link information

**ACTIVITY:** Have participants complete **Activity #3** on the *Activity Sheet*

* The Box Model
* The Box Model concept identifies every element on a page as a rectangular box
* Quickly review the *display* property
  + - Block-level elements occupy the full width of available space and begin a new line
    - Inline-level elements occupy only the width of the content and line up on the same line
    - Inline-block and none are also possible values for the *display* property
* There are several properties that determine the size of a box
* The core of the box is defined by the width and height of an element
* Padding, border and margin expand the dimensions of the box
* Describe the formula for calculating the total size of an element with padding, border and a margin
  + - **Example:**   
      The actual width of an element 400 pixels wide plus a padding (left and right) of 10 pixels, and a margin (left and right) of 20 pixels is 460 pixels wide  
        
      400+10+10+20+20 = 460  
        
      If a left and right border of 3 pixels were added that total would then become 466 pixels wide
* Outline the following steps for setting the width of an area
  + 1. Open
    2. Click the **Save** button

**ACTIVITY:** Have participants complete **Activity #4** on the *Activity Sheet*

* How to position elements
* Define the following
  1. Floats
     + The *float* property was originally designed to allow text to wrap around images
     + It then became a way for taking other elements out of normal page flow
     + It just one of several options for positioning elements on a web page
     + More commonly used to place two elements next to one another
     + The two most popular values for the float property are *left* and *right*
     + A float is typically applied to elements wrapped by a parent element
     + **Example: HMTL** <header>  
        <p>paragraph one</p>  
        <p>paragraph two</p>  
        </header>  **CSS** p {float: left;}  
         
       *The header element is a parent to both paragraph elements. The header element is considered wrapped around both paragraphs therefore contains them. We could apply a float property to each paragraph to force them to appear side by side or on the right or left side within the parent element.*
  2. Clearing Floats
     + The *clear* property is used to prevent content from wrapping around floated elements
     + Using this property will contain the *float* and return the page to its normal flow
     + There are 3 values that can be applied to a float property; *left*, *right*, and *both*
     + The value of *both* will clear left and right floats
     + **Example:** div {clear: left;}
* Demonstrate how the *float* property works

1. Open <http://codepen.io/Meggen/pen/vxELqy?editors=1100>
2. Demonstrate how *float* is used to horizontally align two lists within the footer element
3. Click the **Save** button

**ACTIVITY:** Have participants complete **Activity #5** on the *Activity Sheet*

**Wrap Up/Closing (3 mins.)**

* Ask if there are questions and answer any that were “parked” during the session
* Thank participants for coming and ask them to complete the class survey before leaving