

*CS 641, Haik Sahakian*

---

# Mobile Content Development

---

Readings and Assignments  
Week 2

# Web Development Basics

*Many of you can skip this section, but it'll be helpful to  
anyone who's new to making web pages.*

---

# HTML: Introductory Videos

---

Optional. If you're new to HTML and CSS, this section of this week's assignment is for you. If you're experienced with web development you can skip the next 3 pages.

I think the best way to get up to speed is to watch some short videos, then view more detailed overviews, then check reference sites as needed.

The CSS video will make more sense if you watch the HTML videos first. These videos by Jake Wright are each 12 minutes in length, and are excellent.

HTML: <https://www.youtube.com/watch?v=bWPMSSsVdPk>

HTML part 2: <https://www.youtube.com/watch?v=KJ13lX20FqU>

CSS: <https://www.youtube.com/watch?v=0afZj1G0BIE>

---

# HTML: Detailed Tutorials

---

Optional. Code Academy is a well known and excellent site for learning HTML. However, W3Schools may be much faster, as you can easily skip parts you have already figured out. It's older and not interactive, but for programmers it's sometimes faster you'll learn the basics quickly just by looking at its examples.

W3Schools: [http://www.w3schools.com/html/html\\_intro.asp](http://www.w3schools.com/html/html_intro.asp)

Code Academy: <http://www.codecademy.com/en/tracks/web>

---

# HTML: Reference

---

When you need to look up HTML tags, CSS styles, or JavaScript functions, Mozilla's site below is nicely organized and full of useful information. MSDN and W3Schools are excellent sites too.

HTML: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element#S>

CSS: <https://developer.mozilla.org/en-US/docs/Web/CSS/Reference>

---

# JavaScript: Introductory Video

---

- ❖ *Learn JavaScript in 10 Minutes.* Optional. An introductory video on JavaScript by Jake Wright. If you're comfortable with JavaScript you should skip this video.

[https://www.youtube.com/watch?v=Ukg\\_U3CnJWI&index=5&list=PLlj9BrHKq9WKaz8UV3BjEqicn-C3qHxy4](https://www.youtube.com/watch?v=Ukg_U3CnJWI&index=5&list=PLlj9BrHKq9WKaz8UV3BjEqicn-C3qHxy4)



---

# The Command Line

---

- ❖ *The Mac OS Command Line.* Optional. Mac users new to the Mac Terminal and its commands should read this guide.  
<http://blog.teamtreehouse.com/introduction-to-the-mac-os-x-command-line>
- ❖ *The Windows Command Line.* Optional. Windows users new to the Windows Command Prompt and its commands should read this guide.  
<http://news.softpedia.com/news/basic-guide-for-the-windows-command-prompt-494379.shtml>

# Readings



---

# Responsive Design

---

- ❖ *Responsive Web Design Fundamentals.* An article from Google with an overview of viewports and detailed recommendations for responsive design. Please read all four topics, and come away with an understanding of what the `<meta name="viewport">` tag does on mobile devices. About 30 minutes.

<https://developers.google.com/web/fundamentals/layouts/rwd-fundamentals/>

---

# CSS Grid

---

- ❖ *Building pages with CSS Grid.* Optional. If you feel comfortable with the grid css we used in class, you can skip this. If not, this article is a good introduction to it. 15 minutes.

[https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_Grid\\_Layout/Realizing\\_common\\_layouts\\_using\\_CSS\\_Grid\\_Layout](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout/Realizing_common_layouts_using_CSS_Grid_Layout)

---

# CSS Selectors

---

- ❖ *CSS Selectors*. So far we've touched on just a few CSS selectors: "." for classes, "#" for IDs, no selector for tags, and ":hover" for hovering.
- ❖ There are many more, including content-aware selectors and functions like nth-child().
- ❖ Check out this comprehensive list and explanation of selectors from Mozilla. It's a dense and comprehensive page that you can afford to skim, paying most attention to the examples, and any sections you find interesting. Knowing about advanced selectors will save you time.

[https://developer.mozilla.org/en-US/docs/Web/Guide/CSS/Getting\\_started/Selectors](https://developer.mozilla.org/en-US/docs/Web/Guide/CSS/Getting_started/Selectors)

---

# Recap

---

This is important. Before moving on, make sure you understand what all the following tags and styles do. This list is the *minimum* you need to know, so if any of these are unfamiliar, look them up on the [reference links](#) before doing the rest of the assignments.

- ❖ HTML tags: html, head, body, div, header, footer, nav, section, p, a, button, img, h1-6, style, script, meta, meta name="viewport", link.
- ❖ CSS styles: display, @media, grid-template-columns, grid-auto-columns, grid-gap, margin, padding, min-width, max-width, order, color, background, font.
- ❖ CSS @media queries (*different from styles*): max-width, max-device-width, min-width, min-device-width, orientation.

# Assignments



---

# Assignments

---

- ❖ Ensure you've installed the Chrome browser on your laptop.
- ❖ Chrome currently leads the browsers on developer tools and feature support, and is the most similar desktop browser to mobile browsers. Assignments that do not work on Chrome will not be accepted.



---

# Assignments

---

- ❖ Ensure an image editor has been installed on your laptop. We will be using it in the next class. To check if installation was successful, check you can edit an image from the internet and then save it.

---

# Assignments

---

- ❖ Look at the home pages for Impossible Bureau, Melanie's Look Book, and Apple, to see how they have used areas of flat color, shadow, and color gradients in their design. (30 seconds on each page is fine).
- ❖ On each page, try to identify which areas of the page were styled with pure CSS, and which areas were styled with images. Are any of the gradients actually images? Is any of the text an image? If it's hard to tell, right click on the part of the page you're investigating, and choose "Inspect".

---

# Assignments

---

- ❖ Look at the following three sites on a phone and on a computer. Resize the browser window and see how the layout and content change as the window gets smaller. What happens to the page navigation? Is a separate site necessary for mobile?
- ❖ <http://www.pace.edu/>
- ❖ <http://newlab.com/>
- ❖ <http://lederniergaulois.nouvelles-ecritures.francetv.fr/>

---

# Assignments

---

- ❖ Create a responsive web page for the fictional Ultra Corporation. The page should try to sell the reader something. Give the page a 3-column layout, and make its layout responsive to the width of the browser, so that the page looks good on smaller browsers, and uses a 1-column layout on phones.
- ❖ Use DIVs or other HTML tags styled with grid styles and @media queries to implement your layout. Writing the layout code the first time will take a while. Use @media queries to resize images for phones if needed. On phones, your single column layout will need to hug the edges of the browser window to look good. Switch column widths from pixels to percentages or fractions (fr) on mobile to accomplish this.
- ❖ Use images, links, gradients, rounded corners, shadows, and animation in your page. Don't worry too much about "attractiveness".
- ❖ Test your page on mobile and laptop. Please save the page on [webpage.pace.edu](http://webpage.pace.edu), and post the URL to the Blackboard discussion group for this assignment.

---

# Assignments

---

Your assignment grade will be made up of:

- ❖ 20 points for originality.
- ❖ 20 points for using the html and css features described on the previous page.
- ❖ 10 points for having a working, clickable link posted to your Discussion Group thread rather than just the URL.
- ❖ For 50 points of extra credit, include a 3D transform in CSS.