INTRODUCTION TO: WEB DESIGN

This outline is a work-in-progress, and may change in the future – medatech@medasf.org

MONDAY THROUGH FRIDAY 9AM-1PM @ GOOGLE MEET

MEDA Cohort Fall 2021; Instructor Eduardo Garcia

DESCRIPTION

This is a eight week course that is designed to take a student that has zero knowledge on how to write computer code, and take them to a level that allows them to handle most aspects of building websites. This is an introductory course that will cover a few technologies involved with creating web pages for the Internet. **If you marked absent a total of four days, you are automatically dropped from the class!**

PREREQUISITES

While this is an introductory class to *computer coding*, there are a few things that you should have an understanding of in order to have the best experience in this class. Keep in mind this class does not cover any of these topics during class. You should have a basic understanding of math including addition, subtraction, multiplication, and division. Geometry-level math knowledge is *highly* recommended! The focus of this class is learning how to write code, which means you should feel comfortable using either a Mac or Windows computer and include: opening and saving files, transferring files between computers, using email, and using a web browser.

PENALTIES

- ➔ A total of four days missed from class is grounds for being dropped from the class. Attendance is crucial to your learning. Learning to write code is similar to learning Math, if you miss one day, you might be completely lost the next day! I frown upon tardiness and if you arrive to class after 9:05AM you are marked absent. It is better to be thirty minutes early than thirty minutes late! If you are aware of future absences, please speak to the instructor.
- ➔ A total of three incomplete projects by their respective deadlines is grounds for being dropped from the class. There is no “late work” for this class and the student are responsible to schedule their work on projects outside of class time.

➔ Any type of disrespect toward other individuals or MEDA property may be grounds for being dropped from the class. Any offensive comments or arguments based on racial or religious background toward another person is grounds for being dropped from the class and the student will be asked to leave the class immediately.

(OPTIONAL) RECOMMENDED BOOKS

The following two books are out of date as of 2014 but are still highly recommended for their unique and visual way of showing how code works. Many computer code and programming books are very dry and theoretical but these two books may ease the difficulty into getting started with web design.

➔ HTML and CSS: Design and Build Websites by Jon Duckett

➔ JavaScript and JQuery: Interactive Front-End Web Development by Jon Duckett

CLASS GOALS

- Understand how the web is built.
- The ability to creates web pages that can be seen through any modern web browser, are interactive, and easy to use.
- Have a clear understanding how computer code is written and handled.
- Learn the tools needed to create a website, including Git, GitHub, the Terminal, SSH, and Apache.
- Develop visual design for websites and other media.
- Have a basic skill in writing HTML, CSS, and JavaScript.

WEEK 1

During the first week we will learn how to build a file structure for our web page, how to use the Terminal, Git, and GitHub, and start using the HTML language.

➔ How to name, save, and organize files created especially for your web page or website.

➔ Introduce the software Git and the online service GitHub which is used to “track” your project files.

➔ Learn what a code editor is and why you should use one for writing your code.

➔ We will take a look at the elements of the HTML language to build a basic web page, this includes inserting images, making lists, building tables, and creating forms.

➔ Focus on writing proper code, the importance of tag order, and using tools to validate our code.

WEEK 2

During the second week we will be looking at more advanced topics of HTML, this includes having multiple pages and how linking works, introduce website structure using divs, and a little bit of accessibility.

- ➔ Link web pages and how it compares to linking resources for your web page.
- ➔ Designing a structure for your website using a site map. We will also look at scope creep and how it affects our project and work flow.
- ➔ Designing the structure of each page for easier usability, matching the structure of each page so you don't have CSS nightmares.
- ➔ Why HTML should be "content" oriented.
- ➔ Take a look at the other parts of the HTML document and understand what they are for.
- ➔ Learn how to use the Developer Tools in a browser to further debug your HTML code and understand what the Document Object Model is and how it's different than your code.

For this week's project we will be writing a Hobby web page. Presentations will happen at the end of week.

WEEK 3

During the third week we will be introducing the CSS language which is responsible for the design of a web page.

- ➔ Learn the three different ways to connect our CSS code to our HTML web page, and organizing multiple CSS documents for our website.
- ➔ Look at basic CSS rules that can apply to text, images, buttons, forms, and divs.
- ➔ Understand the "cascading" effect of CSS and learn what applies to this concept. We will also look at "importance" for CSS rules.
- ➔ Switch from the HTML being in control of style, to CSS using CSS classes.
- ➔ How to validate CSS code and avoid pitfalls when having multi-directional relationships between HTML and CSS.

This week's project will be creating a site-map, content, collect images, and outlines for your Portfolio project.

WEEK 4

During the fourth week we will look at a newer technology in CSS for graphics and animation. We will be comparing them to traditional methods and why (while still new) CSS will rule the web. We will also be learning the current method of customizing graphics for our website including re-sizing and optimizing images, avoiding image distortion, and web page resource loading.

- ➔ Look at free tools to help us customize or design graphics for our website.
- ➔ Use Krita for our class and learn basic features of Krita that can be used to develop your website graphics.
- ➔ Use Inkscape to create graphics from scratch and compare raster and vector graphics for their pros and cons.
- ➔ Finding the proper size of an image for your website, and learning why optimizing page load time is important for today's world. Learn more about file and code compression for HTML, CSS, and images.
- ➔ Look at storyboards and wireframes (and wireframe tools) to further organize and present website projects.
- ➔ Introduction to Media Queries and Responsive Design, why it's important and how to use it.
- ➔ Bonus: If we have extra time we will look at CSS Flexbox and Grids.

This week's project will be to take the previous work done and create a Portfolio website. Presentations will happen at the end of week.

WEEK 5

During this week we will be taking a look at additional tools that can give an extra boost to your web design skills. We will also be looking at planning, structuring, designing, and implementing a fictional company's website.

- ➔ Practice using Google Fonts and understanding how it works and why font choice is difficult (for legal reasons) to work with for web projects.
- ➔ Look at a CSS library called FontAwesome, which stores icons in a font to be used on your website (similar to WingDings).
- ➔ Take a look at the basics of Color Theory, tools that can help you develop colors, and take a look at Adobe Color and Flat UI. Look at the Flat UI replacement Material Design, a set of rules set by Google for user experience.
- ➔ Get a head start on the JavaScript language.
- ➔ Further review how to keep our code clean for others (potentially for your teammates) so that it is easier to work with your code. Review indentation and the importance of it.

WEEK 6

This week we will take a look at our first programming language, JavaScript, which is the programming language of the web. We will use JavaScript to add some intelligence into our web pages such as reacting to user's button presses or their mouse movements.

- ➔ Learn what the difference between a computer language and a programming language is.
- ➔ Comparing a programming language to a food recipe.
- ➔ Introduce JSBin to test snippets of JavaScript.
- ➔ Learn how to work with variables, datatypes, arrays, loops, and conditional statements.
- ➔ Look at JavaScript built-in libraries like Math and Date.
- ➔ Understand and debug JavaScript errors.
- ➔ A preview on how to link JavaScript to our web pages and modify the Document Object Model "on-the-fly" with programming logic.

WEEK 7

This week we will wrap up by introducing the jQuery library, the simpler way of using JavaScript with your website, as well as look at how to host websites on a web host.

- ➔ Learn how to add jQuery to our website projects and how it depends on JavaScript to work.
- ➔ Learn and use jQuery syntax to do the same modifications as vanilla JavaScript but with less code.
- ➔ Understand what the Document.ready() function is and why it's important during loading.
- ➔ Carefully analyze JavaScript scopes and closures to avoid common code errors.
- ➔ Take a quick look at jQuery's animation engine, compare it to CSS Animation, and look at jQuery's mini libraries jQuery UI and jQuery Mobile.

This week's project we will be building a website where we create an interactive story, there will be a statement and two buttons, the user can choose which "story path" to take. Presentations will happen at the end of week.

NOTES