Web Languages

Intro

There are a *ton* of different languages used across the web. This module will introduce a handful of them. Broadly speaking there are languages that deal with content and layout, client side interactions and server side interactions. We will talk about the language options for all of these.

Document Structure

HTML stands for Hypertext Markup Language. When people think of the language used to write web pages, this is usually what comes to mind. But we need to be careful with what the purpose of HTML is. It defines the document structure. No more, no less. It is *not* responsible for what the page looks like. Browsers have conventions about how certain HTML elements should look but the job of HTML is not to decide the look of the page. It is to decide the structure and meaning of the content.

Page Look and Feel

CSS stands for Cascading Style Sheets. This is the language that is used to actually modify the layout and styling of a page. HTML might define what is in a particular *section* but CSS will determine what that *section* looks like. CSS can change the spacing between things, the color of things or even make very simple interactivity possible. For example is is possible to change the color of something when you mouse over it using CSS. However there is not much more than that in terms of logic.

Client Side Interactivity

When we talk about client side interactions we are pretty much exclusively talking about *JavaScript*. JavaScript is a language that has syntax that resembles the C language. It has all of the control structures one would expect to find in a robust programming language. However, the type and class system is going to be different than what you are used to.

We will be doing *a lot* of work with JavaScript in this class. It has a host of built in features that let it do things like interact with the structure of a page, change the styling of a page and make requests to servers to get data. One can have simple JavaScript that just changes the look here and there or entire fluid dynamics simulations can be written in JavaScript and rendered in the browser.

Server Side Languages

So far it has seemed pretty clear. One task, one language. Server side is where it all goes off the rails. There are a huge number of languages that can be used, often with different parts of bigger servers written in different languages. *PHP, C#, PHP, JavaScript, Ruby, Go, Python* - These are just a few of the many different languages servers are programmed in. Some, like Python or JavaScript, are seldom used alone but are instead used with a framework for server side programming. In this class we will be using Node.js and JavaScript to program on the server. Django is a very popular framework that is used with Python. However some languages like PHP do not require a framework at all and have many of those features baked into the language itself.

Activity

No activities for this section. This is really just some background material to give some context.

Review

As we learned previously there are different layers to a web page and several stages in a pages life-cycle. This section detailed some of the most important languages involved in those different layers. HTML and CSS are both used by the browser to structure a page and style it. JavaScript it made to make that page respond to client side events and any one of a very large selection of languages can be used on the server to generate pages with dynamic content.