

CS50 Week 9

Sam Green '17

samuelgreen@college.harvard.edu

646-457-2340

Today's Agenda

JavaScript: Basics & Use Cases

DOM – Document Object Model

Practice with JS and the DOM

A Toy API Example

PSet

What is JavaScript?

Client-side dynamic programming language

Speed and easy of use from client side language.

JS doesn't fit into one category of programming

Has elements of functional, object-oriented, and imperative programming

Interpreted, not compiled (like PHP)

Can manipulate the DOM to change HTML – one of the reasons it is so popular.

Functions as Values

Functions in JavaScript are values.

They can be assigned to variables.

They can be passed as arguments or returned as results from other function calls.

Anonymous Functions

```
var f = function(){  
    alert("Sam!!!!");  
};  
f();
```

Objects & Functions as Values

An object is a container for related values.
Some of these values can be functions (!).

Example: JS onsubmit function from lecture.

Example: JS onclick

Document Object Model (DOM)

Hierarchical way of laying out information

HTML webpage is an example.

Let's draw a DOM tree for

```
<!DOCTYPE html>
  <head>
    <title id="title">Hey Sam</title>
  </head>
  <body>
    <div>
      
    </div>
  </body>
</html>
```

JS and the DOM

JS allows us to dynamically alter the DOM,
without reloading the page.

Key point on this: we can use JS to select
elements from the DOM and then change them.

JS Validation

What are the benefits of client-side form validation?

What can we validate?

Is the PHP back-end still necessary?

Let's do some DOM
manipulation examples.

Ajax and Asynchronous Requests

Ajax is an JavaScript paradigm for asynchronous web requests.

Ajax is let's webapps be powerful.

Example in life: when new tweets appear on your Twitter feed, that happens via Ajax.