# Intro to JavaScript

ICS4U - Mr. Emmell

## What is JavaScript?

- JavaScript is a programming language for use in HTML pages
- Invented in 1995 at Netscape Corporation
  - (Called LiveScript at the time)
- JavaScript has very little to do with the Java programming language
- JavaScript programs are run by an interpreter built into the user's web browser (not on the server)

# Where does JavaScript fit in?

- Recall
  - 1. client opens connection to server
  - 2. client sends request to server
  - 3. server sends response to client
  - 4. client and server close connection
- What about Step 5?
  - 5. Client renders (displays) the response received from server
- Step 5 involves displaying HTML and running any JavaScript code within the HTML

# What can JavaScript do?

- JavaScript can dynamically modify an HTML page
- JavaScript can react to user input
- JavaScript can validate user input
- JavaScript can be used to create cookies (yum!)
- JavaScript is a full-featured programming language
- JavaScript user interaction does not require any communication with the server

# Using JavaScript in your HTML

- JavaScript can be inserted into documents by using the SCRIPT tag

```html
<html>
  <head>
    <title>Hello World in JavaScript</title>
  </head>
  <body>
    <script type="text/javascript">
      document.write("Hello World!");
    </script>
  </body>
</html>
```

# Where can you put your scripts?

- You can have any number of scripts
- Scripts can be placed in the HEAD or in the BODY
  - In the HEAD, scripts are run before the page is displayed
  - In the BODY, scripts are run as the page is displayed
  - At the end of the BODY, scripts are run after the page is displayed
- In practice, JS is usually written at the end of the body

# Using JavaScript in your HTML

```html
<html>
  <head>
    <title>Hello World in JavaScript</title>
    <script type="text/javascript">
      function helloWorld() {
        document.write("Hello World!");
      }
    </script>
  </head>
  <body>
    <script type="text/javascript">
      helloWorld();
    </script>
  </body>
</html>
```

# External Scripts

● Scripts can also be loaded from an external file
● This is useful if you have a complicated script or set of subroutines that are used in several different documents

<script src="myscript.js"></script>

# JavaScript Variables

- JavaScript has variables that you can declare with the optional var keyword
- Variables declared within a function are local to that function
- Variables declared outside of any function are global variables

var myName = "Mr. Emmell";

# JavaScript Operators & Constructs

- JavaScript has most of the operators we're used to from C/Java
  - Arithmetic (+, - , *, /, %)
  - Assignment (=, +=, -=, *= /=, %=, ++, --)
  - Logical (&&, ||, !)
  - Comparison (<, >, <=, >=, ==)
- Note: + also does string concatenation
- Constructs:
  - if, else, while, for, switch, case

# Simple User Interaction

• There are three built-in methods of doing simple user interaction

– alert(msg) alerts the user that something has happened

– confirm(msg) asks the user to confirm (or cancel) something

– prompt(msg, default) asks the user to enter some text

```
alert("There's a monster on the wing!");
confirm("Are you sure you want to do that?");
prompt("Enter you name", "Adam");
```

# JavaScript Functions

● JavaScript lets you define functions using the function keyword
● Functions can return values using the return keyword

```
function showConfirm() {
  confirm("Are you sure you want to do that?");
}
```

# Comments in JavaScript

- Comments in JavaScript are delimited the same as C
  - Block comments with  /*    */
  - Single line comments with  //

# JavaScript Arrays

- JavaScript has arrays that are indexed starting at 0
- Special version of for works with arrays

```
<script type="text/javascript">
  var colors = new Array();
  colors[0] = "red"; colors[1] = "green";
  colors[2] = "blue"; colors[3] = "orange";
  colors[4] = "magenta"; colors[5] = "cyan";
  for (var i in colors) {
    document.write("<div style=\"background-color:"
                      + colors[i] + ";\">"
                      + colors[i] + "</div>\n");
  }
</script>
```

# JavaScript Events

- JavaScript can be made to respond to user events
- Common Events:
  - onclick : when the object is clicked
  - onload and onunload : when a page is first visited or left
  - onfocus, onblur, onchange : events pertaining to form elements
  - onsubmit : when a form is submitted
  - onmouseover, onmouseout : for "menu effects"
- A complete list of event types is available here
  - https://www.w3schools.com/jsref/dom_obj_event.asp

# JavaScript and the DOM

- The Document Object Model (DOM) is a specification that determines a mapping between programming language objects and the elements of an HTML document
- Not specific to JavaScript
- The Document object represents an HTML document and can be used to access all objects in a page

# HTML DOM - The Document Tree
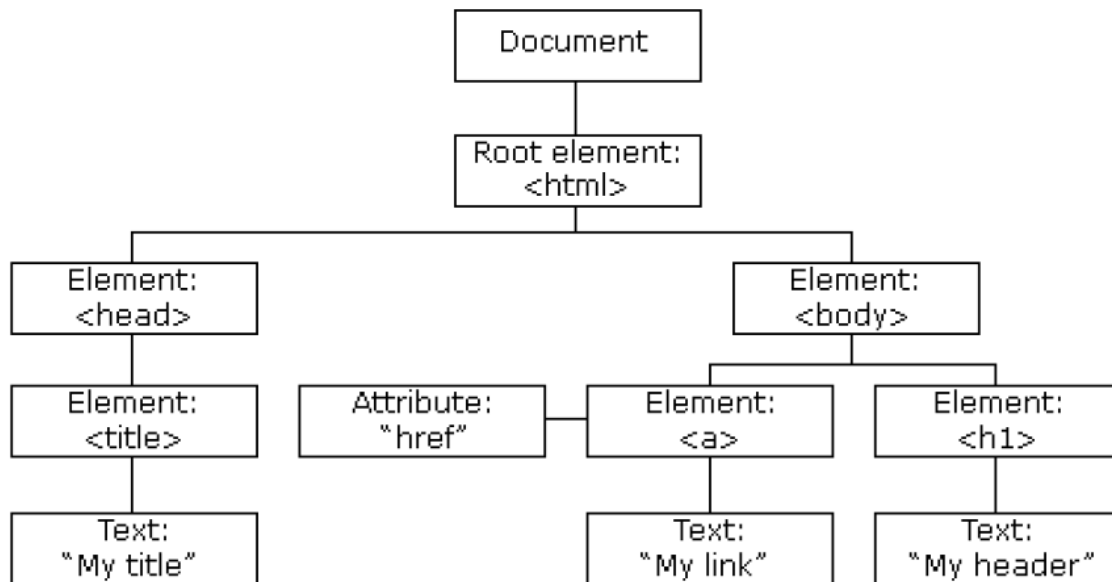


# HTML DOM Objects

- Environment objects
    - Window, Navigator, Screen, History, Location, Document
- HTML objects
    - Anchor, Area, Base, Body, Button, Event, Form, Frame, Frameset, Iframe, Image, Checkbox, FileUpload, Hidden, Password, Radio, Reset, Submit, Text, Link, Meta, Object, Option, Select, Style, Table, TableCell, TableRow, TextArea
- The document itself has several useful methods
    - getElementById, getElementsByName, getElementsByTagName, write, writeln, open, close

# So how do we use it?

● Simplest approach is to make a variable point to an object, then change it.

```html
<input type="text" id="myTextField"> <br><br>
<button onclick="changeTheText();">Click me</button>

<script type="text/javascript">
  function changeTheText() {
    var target = document.getElementById("myTextField");
    target.value="A whole new world!";
  }
</script>
```

# But it's not working!

All errors are reported in the console ONLY!

You MUST be looking at the console when you are writing JavaScript code

Right click → Inspect → Console Tab