Lecture One: Language & Scripting Basics

What JavaScript Is and Isn't

- JavaScript is a programming language that you can use to add interactivity to your web pages
- A JavaScript script is a program this is included on an `HTML` page
- Because it is enclosed in a `<script>` tag, the text of the script does not appear on the user's screen
- The web browser knows to run the JavaScript program
- The <script> tag is most often found in the `<head>` tag, though you can put scripts in the `<body>` tag (i.e., script that write text on the screen) – ***p. 2, Script 1.1***
- Despite the name, JavaScript and Java have almost nothing to do with one another
- Java is a full featured programming language developed and marketed by Sun Microsystems
- With Java, a descendant of the C and C++ programming languages, programmers can create applications called applets that are small programs that download over the Internet and run in browser windows

What JavaScript Can Do

- JavaScript lets you create an active user interface, giving users feedback as they navigate thru your pages
- Buttons that highlight as you mouse over them called rollover buttons
- You can make sure that your users enter valid information in forms
- With JavaScript you have the ability to create custom `HTML` pages
- You can have new windows open up, display alert boxes and put custom messages in the status bar
- JavaScript has a set of date and time features like clocks, calendars and timestamp documents

Language Basics

Objects – *p. 11*

- JavaScript is an ***object*** oriented language
- An object is some kind of thing such as a *window*, a *form*, etc.
- Because you can have more than one object, it makes sense to give the object their own unique names
- Objects can also have sub-objects

Properties – *p. 12*

- Objects have ***properties***
- In the JavaScript world, a window has a *title*, and a form can have a *checkbox*
- Changing a property of an object can modify the object, and the same property can apply to completely different objects

Values & Value Types – *p. 15*

- a ***value*** is a piece of information In JavaScript, the kind of value you may be most familiar with are ***numerical values – p. 15, Table 1.2***

- A ***string value*** is a sequence of letters, digits, punctuation characters enclosed within single or double quotation marks used to represent text, parameters and data

Variables – *p. 15*

- A ***variable*** is a name associated with a value that stores that value for it to be used in the JavaScript
- A ***variable*** must be declared within your JavaScript in order to use that variable within the equation

Operators – *p. 15*

- ***Operators*** are the symbols used to work with values and variables – ***p. 15, Table 1.3***
- You are already familiar with operators from simple arithmetic like Plus, Minus and Equals

Assignments – *p. 16*

- When you give a variable a value, you are ***assigning*** that value to the variable and you use an ***assignment operator*** to do the job – ***p. 16, Table 1.4***

Comparisons – *p. 16*

- You will often want to ***compare*** the value of one variable with another, or the value of a variable against a literal value – ***p. 16, Table 1.5***
- ***Boolean values*** are generally the result of ***comparisons*** you make in your JavaScript code and has only two possible values: True or False

Scripting Basics

Where to put your Scripts – *p. 23*

- Scripts can be put in one of two places on an HTML page
- A “Header Script” gets put between the `<head>` and `</head>` tags
- A “Body Script” gets put between the `<body>` and `</body>` tags

The Opening Script Tag – *p. 23, Script 1.1*

```
<script language=”Javascript” type=”text/javascript”>
```

- The `<script>` tag tells the browser to expect JavaScript instead of HTML
- The `language=”Javascript”` attribute identifies to the browser which scripting language is being used
- The `type=”text/javascript”` attribute tells the browser that the script is plain text, organized as java script

Hiding Scripts from Older Browsers

- Very old browsers, which include Netscape 1.x, Microsoft Internet Explorer versions 3 and earlier and the AOL browser before version 4 do not understand JavaScript
- Most browsers are supposed to ignore everything between tags that they do not understand, however, not all of them do
- There is a way to trick these browsers to make them think that the contents of your script are HTML comments which they will then ignore

```
<script language="Javascript" type="text/javascript">
<!– Hide  script  from old browsers

   document.write("Hello, world!")

-->
</script>
```

- Any text inside these special tags gets ignored by older browsers – notes, clarifications, explanations, etc.
- This is not so important these days, however it is still good practice

Putting Comments in Scripts – *p.29, Script 2.4*

- It is a good idea to get into the habit of adding comments to your scripts
- Comments will help to explain why you solved the problem in a particular way
- Other people who need to re-use or modify your script will benefit from comments
- You can do this by inserting comments that JavaScript won't interpret as script commands

```
<script language="Javascript" type="text/javascript">
<!-- Hide script from old browsers

// this is a one line comment without returns

/*  This is an example of a multi-line JavaScript comment.

It has a hard return

Note the characters at the beginning and ending of the comment. */

   document.write("Hello, world!")

-->
</script>
```

Creating an Alert

- Alerts are used in a wide range of applications
- A simple alert uses a predefined function called a **method** with a **string value** of text

```
<script language="JavaScript" type="text/javascript">
<!-- Hide from old browsers

   alert("Welcome to MIC102G - Introduction to programming with
JavaScript!")

-->
</script>

// within the script tags, use the pre-defined “alert()” method

// since we are putting the “alert()” method within the script tag, the
browser will read the script tag and then read the method.

// as the browser reads the method it will perform the function of
opening an alert box before the browser can read thru to the body tag
```