Introduction to JavaScript

CST 365 – Web Applications
Michael Ruth, Ph.D.
Assistant Professor
Computer Science & I.T.
mruth@roosevelt.edu

Objectives

- Discuss the purpose and current use of JavaScript and its relationship with Java
- Describe how and where JavaScript can be used in conjunction with HTML
- Discuss learning to programming in JavaScript
- Discussed DOM as it relates to JavaScript and its use to manipulate Web pages directly
- Explain several DOM objects including their purpose and their use and discuss events
- Provide examples of JavaScript Programming

What is JavaScript?

- JavaScript was designed to add interactivity to HTML pages
- JavaScript is a scripting language
  - A scripting language is a lightweight programming language
- JavaScript is usually embedded directly into HTML pages
- JavaScript is an interpreted language (means that scripts execute without preliminary compilation)
- Everyone can use JavaScript without purchasing a license

What can it do?

- Javascript serves to make Web pages dynamic rather than static
  - Can be used to detect the browser type
  - Can put dynamic text onto the page
  - Can manipulate (read and write) HTML elements

What is it used for in practice?

- Validate form data automatically
  - Current Best Practice!
- Used to manipulate (read/write) cookies!
  - Cookies are used to maintain state information
- React to events
  - Browser or user events

JavaScript/Java/EMCAScript

- Java is a generic programming language
  - Is compiled into bytecode
- JavaScript is a scripting language
  - Is designed to interact with the browser
- JavaScript is officially named "EMCAScript"
  - JavaScript got its name because of a marketing ploy by Netscape to give JavaScript the cachet of what was then the hot new web-programming language

How do I use it?

- JavaScript is embedded into the HTML document by means of the <script> tag
  - Type attribute must be set to "text/javascript"
  - Ex:

```
<script type="text/javascript">
        document.write("Hello World!");
</script>
```


Browser Problems?

- A browser normally ignores unknown tags.
  - This means that an old browser that does not support scripting, will ignore the <script> tag, but the content of the tag will be displayed on the page
  - To avoid this:

```
<script type="text/javascript">
    <!–
        document.write("Hello World");
    -->
</script>
```


Where do I use it?

- Javascript can be inserted in the Body or in the Head
  - <head>
    - the script waits to be called (functions)
      - Used for functions and event handling
  - <body>
    - The script executes when the page is loading

JavaScript as a PL

- Comments
- Output
- Syntax Comments
- Variables (Text/Data)
- Expressions/Operators
- Conditional Statements & Loops
- Functions
- Important Objects I

JavaScript Comments

- The syntax for a single-line comment is:
  // *comment text*
- The syntax of a multi-line comment is:
  /*
  comment text covering several lines
  ***/**

Output

- JavaScript provides two methods to write text to a Web page:
  - **document.write("text");**
  - **document.writeln("text");**
- They differ exactly as you expect them to!

Syntax Comments

- JavaScript commands and names are case-sensitive
- JavaScript command lines end with a semicolon to separate it from the next command line in the program.
  - in some situations, the semicolon is optional
  - semicolons are useful to make your code easier to follow and interpret

Variables

- Named locations in memory
  - Naming conventions are same as they are in java (KISS)
  - Names are case-sensitive
- Declaring variables uses var keyword
  - **`var variable;`**
    - Note: no real type is specified…

Variable Types

- JavaScript supports four different types of variables:
  - numeric variables can be a number, such as 13, 22.5, or -3.14159
  - strings are any group of characters, such as “Hello” or “Happy Holidays!”
  - booleans accept one of two values, T or F
  - null is a variable that has no value at all

Operators

- All the familiar operators are here in the precedence you are familiar with
  - Arithmetic operators are for numbers, but + is concatenation for strings
- Comparison operators are also here

Branching/Looping

- Branching (if statements, if/else, etc) are the same as they are for java
- Looping statements are also the same as they are in java
  - for, while, do-while all included

Functions

- A function is a method defined by using the **function** keyword

```
function function_name(parameters) {
… function body
}
```

Functions (Cont.)

- Calling a function with arguments:

```
function myFunction(var1,var2)
{
 … function code …
}
```


Functions (Cont.)

- Returning a Value:

```
function myFunction()
{
var x=5;
return x;
}
```


JavaScript & Objects

- JavaScript is an OOP
  - We will rarely implement our own objects
  - However, there are some VERY important objects to learn how to use!
  - Some of which we will talk about today, and some we will leave for another day!

The String Object

- Declaring String variable:
  - var txt = “My String”;
- String properties:
  - length -> returns the length of the string
- String methods:
  - charAt(), substr(), toLowerCase(), etc!
  - We’re familiar with most of these!!!

Other Built-in Objects

- Date
  - Class used to get/manipulate dates
- Math
  - Class with math utility functions
- Array
  - The Array object is used to store multiple values in a single variable

DOM Objects

- What it stands for:
  - DOM => Document Object Model
- What it means for us:
  - A set of objects which will allow us to interact with the browser and web page

Basic DOM Objects

- Window Object (Most Important)
  - Refers to the browser window!
- Navigator Object:
  - contains information about the browser
- Screen
  - Contains information about users screen
- History
  - Contains information about user's history
- Location
  - contains information about the current URL
  - Actually part of Window object

Document Object

- Each HTML document loaded into a browser window
- Provides access to all HTML elements in a page, from within a script
- From this class, we can get:
  - All the images, forms, links, etc.
    - A bit easier if we name/ID things!

Document Attributes/Methods

- Collections (arrays)
  - links
  - images
  - anchors
  - forms
- Very important:
  - getElementByName()
  - getElementByID()

Events

- Events are actions that can be detected by JavaScript
  - Since JavaScript can view the browser & the page, there are a lot of involved events
  - Events are normally used in combination with functions
- There are a ton, but we'll look at a couple!

Important Events

- Mouse Movements:
  - onclick, click, mouseover, etc.
- Keyboard Movements:
  - keypress, keyup, etc.
- Document/Browser Events:
  - Submit, load, etc.

Assigning Events

- Two means of accomplishing this:
  - **Inline**
    - To assign events to HTML elements you can use the event attributes
    - For instance: Assign an onclick event to a button element:
      - <button id="myBtn" **onclick="displayDate()"**>Try it</button>
  - **DOM**
    - You can assign events to HTML elements using DOM as well:
    - For instance: Assign an onclick event to a button element:
      - document.getElementById("myBtn").onclick=function(){displayDate()};

For More Information

- W3 Schools (great tool for studying)
  - HTML
    - http://www.w3schools.com/html/default.asp
  - CSS
    - http://www.w3schools.com/css/default.asp
  - JavaScript
    - http://www.w3schools.com/js/default.asp

Summary

- Discussed the purpose and current use of JavaScript and its relationship with Java
- Described how and where JavaScript can be used in conjunction with HTML
- Discussed the fundamentals of programming in JavaScript
- Discussed DOM as it relates to JavaScript and its use to manipulate Web pages
- Explained several DOM objects including their purpose and their use and discussed events
- Provided examples of JavaScript Programming

Questions?