

# Session 6



## JavaScript Part 1

# Reading

## ■ Reading

### ■ Wikipedia

[en.wikipedia.org/wiki/Javascript](http://en.wikipedia.org/wiki/Javascript)

### ■ Web Developers Notes

[www.webdevelopersnotes.com/tutorials/javascript/](http://www.webdevelopersnotes.com/tutorials/javascript/)

### ■ JavaScript Debugging

[www.w3schools.com/js/js\\_debugging.asp](http://www.w3schools.com/js/js_debugging.asp)

## ■ jQuery-DOM

<http://www.ibm.com/developerworks/xml/tutorials/x-processxmljqerytut/index.html>

*We cover jQuery later in the  
course*

## References

### ■ Reference

#### ■ ECMAScript

[www.ecma-international.org/publications/files/ECMA-ST/ECMA-262.pdf](http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-262.pdf)

#### ■ Mozilla Guide

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference>

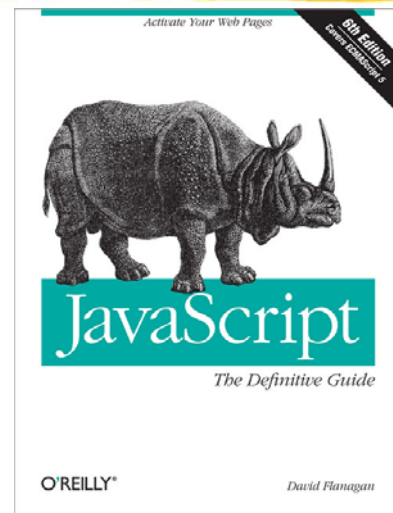
#### ■ W3C

[www.w3.org/TR/REC-html40/interact/scripts.html](http://www.w3.org/TR/REC-html40/interact/scripts.html)

## Reference Book

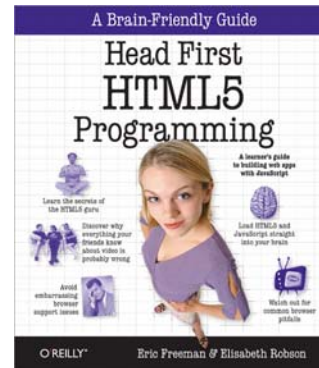
- JavaScript: The Definitive Guide by David Flanagan, O'Reilly Press, 6<sup>th</sup> Edition (might be available through Safari Books On-Line and Google Books)

the only complete JavaScript  
book I have seen that is  
written from a CS  
perspective



## Background Reading

- Also possibly available through CS Library using Safari Books On-line
  - Head First HTML5 Programming: Building Web Apps with JavaScript by Eric Freeman and Elisabeth Robson, O'Reilly Press
  - XML In a Nutshell, Chapter 19 (DOM)



Not written for CS majors,  
but reasonably correct

© Robert Kelly, 2018

5

## Learning Goals

- Understand differences of JavaScript as compared with Java
- Understand syntactic and semantic structure of JavaScript
- Understand use of events

© Robert Kelly, 2018

6

## What is JavaScript?

- A scripting language (i.e., a lightweight programming language) to use within a browser
- Usually embedded directly into HTML pages
- The sole surviving language for Web client programming
- An interpreted language (means that scripts execute without preliminary compilation)

The name officially refers  
to ECMAScript

© Robert Kelly, 2018

7

## Why Should You Learn JavaScript

- Useful for
  - Client side form processing (e.g., field validation)
  - More dynamic graphic UI
  - Dynamic update of html pages - Ajax

© Robert Kelly, 2018

8

## Ajax

- JavaScript is essential to the use of Ajax
- Ajax provides a new Web interaction style
- Examples:

[maps.google.com/maps](https://maps.google.com/maps)

[nyc.bestparking.com/](https://nyc.bestparking.com/)

With an Ajax application, parts (but not all) of a page will change based on user input

© Robert Kelly, 2018

9

## Important Concepts

- Low-level syntax of JavaScript is similar to Java, but the object model is very different
- A JavaScript can be set to respond to GUI events
- JavaScript treats functions as first class objects (you can use them in places where you would use other objects)
- JavaScript is a weakly typed language (implicit type conversion)
- **Browsers provide access to the document tree with JavaScript using the Document Object Model (DOM)**
- JavaScript code can request data from the server - for update of the document tree (and browser update of the page)
- The Browser Object Model (BOM) forms a hierarchy of objects that interact with the browser

© Robert Kelly, 2018

10

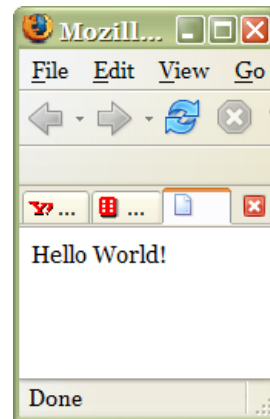
## Hello JS

```
<html>
<body>
<h2>
<script>
  document.write("Hello World!");
</script>
</h2>
</body>
</html>
```

Script tag is used to insert JavaScript into a page

Semicolon is optional (but mandatory for multiple statements on a line)

Code within a script element is executed immediately when the page is loaded (if it is not in a function)



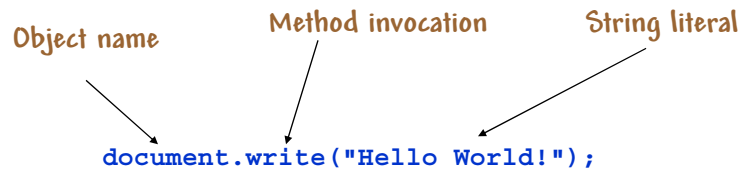
## JavaScript Development

- Major browsers have JavaScript debuggers available
  - Firefox
  - Chrome

More on JavaScript debugging once we cover libraries
- NetBeans has good syntax analysis features

Be careful in debugging - sometimes a JavaScript function will just return if it encounters an error

## JavaScript Object Notation



- Syntax similar to Java
- But what is the document object?

© Robert Kelly, 2018

13

## Window as Global Execution Context

- The document object represents the html document
- The window object represents the browser window that displays the document
- The window object is the global object (think of it as the default object)
- The document object is a property of the window object

`window.document.write (...)`      ↔      `document.write (...)`

Is similar to

`this.getServletContext (...)`      ↔      `getServletContext (...)`

© Robert Kelly, 2018

14

## Script Tag - Type Attribute

### ■ Possible values

- text/javascript
- text/ecmascript - A standard version of Javascript
- text/jscript - Microsoft's version of Javascript
- text/vbscript - Run only in IE
- text/vbs
- text/xml

`<script type="text/javascript">`

### ■ With HTML 5, all you need is

`<script>`

© Robert Kelly, 2018

15

## JavaScript Functions

- Scripts that appear in the head element of the document are loaded first
- A non-function script in the head element will execute before the page loads (not too useful)
- A function defined in the head element will be loaded before anyone uses it, and so is available to any function call in JavaScript located in body

© Robert Kelly, 2018

16



## JavaScript Variables

### ■ Syntax

```
var strname = "x"
```

or

```
strname = "x"
```

- Variables declared within a function are local to the function
- Variables declared outside a function are properties of the window object (visible everywhere in the page)

© Robert Kelly, 2018

17

## Operations

- Arithmetic
- Assignment
- Comparison
- Logical
- String
- Conditional

Syntax is very similar to Java  
(both are based on C)

All Java keywords are  
reserved in JavaScript

© Robert Kelly, 2018

18

## Popup Boxes

- Alert box - user has to click OK to proceed
- Confirm box - user has to either click OK or cancel to proceed
- Prompt box - user enters a value, then clicks either OK or Cancel to proceed

Methods of the Window object

```
alert("Email must be filled out");  
confirm("sometext");  
prompt("sometext", "defaultvalue");
```

Popup boxes act as breakpoints for debugging



© Robert Kelly, 2018

19

## Conditional Statements

- If, else statements

End of statement semicolons can be omitted if each statement is on a separate line

```
<script>  
//If the time is less than 10,  
//you will get a "Good morning" greeting.  
//Otherwise you will get a "Good day" greeting.  
var d = new Date()  
var time = d.getHours()  
if (time < 10) {  
    document.write("Good morning!") }  
else { document.write("Good day!") }  
}  
</script>
```

© Robert Kelly, 2018

20

## External JavaScripts

- Similar to style sheets
  - Script can either be embedded or referenced in an external file

```
<html>
  <head>
    <script src="xxx.js"></script>
  </head>
  <body>
  </body>
</html>
```

© Robert Kelly, 2018

21

## Functions

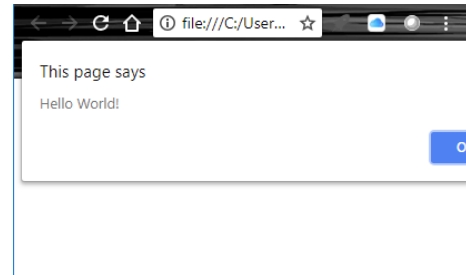
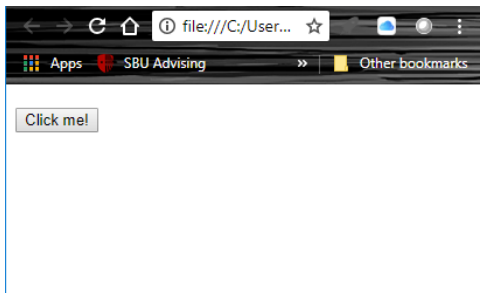
- To keep the browser from executing a script as soon as the page is loaded, write your script as a function.
- A function contains some code that will be executed only by an event or by a call to that function.
- You may call a function from anywhere within the page
- Functions are defined at the beginning of a page, in the <head> section (so that they are available when your page begins to load)

© Robert Kelly, 2018

22

## Function Example

```
<html> <head>  
<script> function displaymessage() { alert("Hello World!") }  
</script> </head>  
<body> <form>  
<input type="button" value="Click me!" onclick="displaymessage()" >  
</form> </body> </html>
```



© Robert Kelly, 2018

23

## Events

- Events are actions that can be detected by JavaScript
- Elements on a Web page have events that can be used to execute JavaScript functions

### ■ Examples

- Mouse click
- Image load
- Mouse over
- Form submittal

```
<input type="button"  
value="Click me!"  
onclick="displaymessage()" >
```

Notice that these event attributes are not camel case

© Robert Kelly, 2018

24

## Typical Event Handlers

- **onclick** - when the pointing device button is clicked over an element
- **onmouseover** - when the pointing device is moved onto an element
- **onchange** - when a control loses the input focus and its value has been modified since gaining focus
- **onblur** - when an element loses focus either by the pointing device or by tabbing navigation
- **onfocus** - when an element receives focus either by the pointing device or by tabbing navigation
- **onsubmit** - when the submit button of a form element is clicked

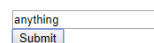
© Robert Kelly, 2018

25

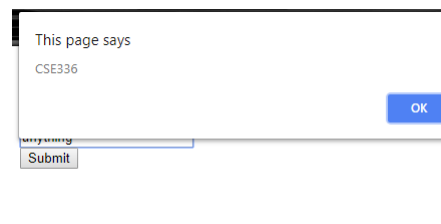
## Are We on Track

- Write an html page that contains a form with a text box and a submit button
- When the text box loses focus (after you enter text and hit tab) display an alert box with the text "CSE336"

### Track - JavaScript Example



anything  
Submit



This page says  
CSE336  
OK

Value of action attribute in form tag does not matter

© Robert Kelly, 2018

26

## Were We on Track?

```
...  
<script>  
function f() { alert("CSE336"); }  
</script>  
</head>  
  
<body>  
<h1>Track - JavaScript Example</h1>  
<form action="http://localhost:8080/CodeCSE336/JSPs/FormTester3.jsp"  
  method="post" >  
<input type="text" name="JS-Input" value="Enter text" onblur="f();" />  
<br /><input type="submit" />  
</form></body></html>
```

© Robert Kelly, 2018

27

## Guidelines

- White space is ignored
- Case sensitive
- Comments (// ...)

© Robert Kelly, 2018

28

## Objects

- Object properties are accessed with the dot (.) operator  
*Object encapsulation practice is not like Java*
- Object methods invoked with the dot (.) operator and a parameter list ( )
- Built-in objects  
*Objects are really maps, where the map value can be a function*
  - String
  - Date
  - Array
  - Boolean

```
var myDate=new Date()  
myDate.setFullYear(2010,0,14)  
  
var mycars=new Array()  
mycars[0]="Saab"  
mycars[1]="Volvo"  
mycars[2]="BMW"
```

© Robert Kelly, 2018

29

## Have You Satisfied the Lecture Objectives?

- Understand differences of JavaScript as compared with Java
- Understand syntactic and semantic structure of JavaScript
- Understand use of events

© Robert Kelly, 2018

30