CO456 / CO458
Web

- most materials adapted from ***Moseley (2007)***, Chapter 6 –
supplemented with extracts from Bates (2006), w3schools.com and McGrath (2006)

Lecture 7

JavaScript ***objects*** & ***the DOM***

Richard Mather and Carlo Lusuardi

Module schedule

Wk.	Lecture/subject area(s)	Practical	Reading (Moseley, 2007)
1	Introduction How the Web works	Internet/Web definitions and HTML report	Ch 1 (The way the Web works)
2	HTML 1 (Introductory - inc. lists and hyperlinks)	HTML	Ch 2 pp 24-36 (HTML)
3	HTML 2 (inc. tables, images and forms)	HTML	Ch 2 pp 36-48 (HTML) Ch 3 (XHTML and frames)
4	CSS 1 (Introduction and core CSS principles)	CSS – introductory styles, embedded styles.	Ch 4 pp 76-96.
5	CSS 2 (Positioning elements).	CSS– using IDs, classes and layout control.	Ch 4 pp 97-103.
6	CSS 3 (Advanced layout & navigation)	CSS – using CSS to produce button-like navigation from HTML list elements. (CW2a to be demonstrated).	Specialised articles.
7	JavaScript 1 (Fundamentals, variables)	JS – foundation constructs.	Ch 5 pp 108-116
8	JavaScript 2 (Functions, branches, loops).	JS – calling functions.	Ch 5 pp 117-124.
9	JavaScript 3 (Objects and the DOM).	JS – manipulating the DOM.	Ch 6 126-139.
10	JavaScript 4 (Forms and validation). And DHTML	JS– validating user completed forms.	Ch 6 139-145, Ch 7.
11	HTML***5***, CSS***3***, - media, forms, gradients, SVG ('Edge') and other enhancements	Web frameworks taster session 1	See practical sheets for information sources
12	Advanced HTML5, CSS3 & JS frameworks (e.g. jQuery, jQuery Mobile, Box2DWeb)	Web frameworks taster session 2	See practical sheets for information sources
	Vacation		
13	Assignment workshop 1	Assignment workshop 1	N/A
14	Assignment workshop 2	Assignment workshop 2	N/A

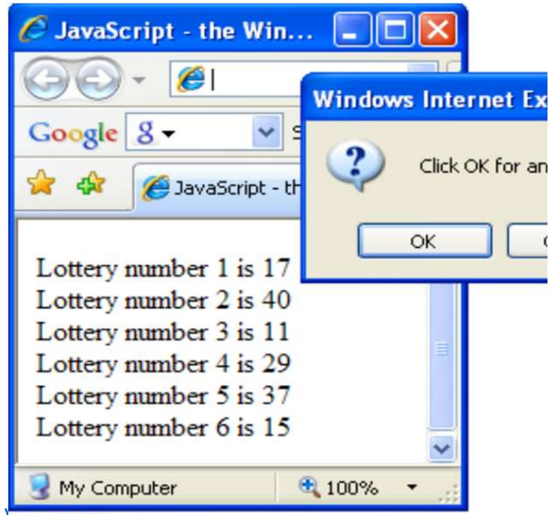
JavaScript Objects

- Predefined / Internal Objects
- Objects in JavaScript include: String, Array, Math, Date
- Objects seen as attempt at more coder-friendly programming
- Can also be seen a group of variables with respective values
- Objects have associated actions or behaviours
- https://www.w3schools.com/js/js_objects.asp

- “An object is a collection of named values [*a.k.a. properties/fields*] and associated methods” (Moseley, 2006)

Objects - Syntax

- myValue [opt] = objectRef.propertyName
- myValue [opt] = objectRef.methodName(parameters [opt])
- https://www.w3schools.com/jsref/
  - – document.write(Math.PI);
  - – myValue = Math.round(10.2);



```
Lottery number 1 is 17
Lottery number 2 is 40
Lottery number 3 is 11
Lottery number 4 is 29
Lottery number 5 is 37
Lottery number 6 is 15
```

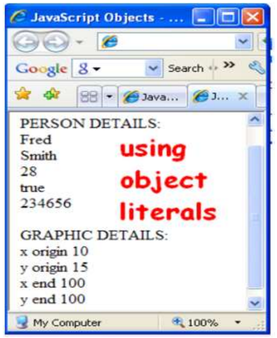
The lottery number example uses two Math methods

- lotteryNumbers[i] = Math.round(Math.random()*49)
- How might this be improved to: [1] output selections in ascending order; and [2] ensure all numbers are unique?

For more Math object features see Moseley (2006) p127, Table 6-1

JavaScript User-Defined Objects

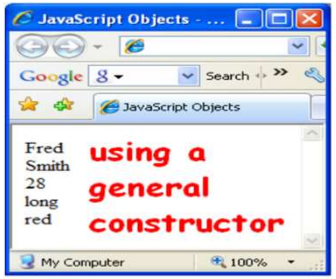
- Objects may be created implicitly using ***object literal values***
  - var person = { firstName: "Fred", lastName: "Smith", age: 28, married: true, telephone: 234656 }



PERSON DETAILS:
Fred
Smith
28
true
234656
GRAPHIC DETAILS:
x origin 10
y origin 15
x end 100
y end 100

using object literals

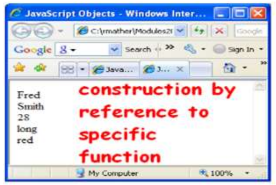
- Objects may be created using a ***general object constructor***
  - General rule is … var myObject = new Object();
  - In the example: var person = new Object();



Fred
Smith
28
long
red

using a general constructor

- Objects may be created directly with ***specific reference*** to a function “pseudoClass”
  - In the example: function person(nm1, nm2, age, lgth, col)



Fred
Smith
28
long
red

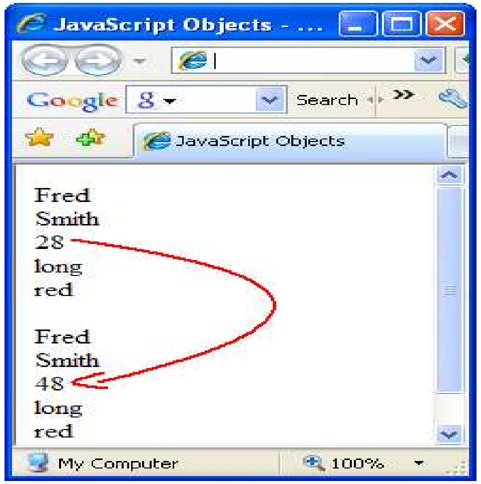
construction by reference to specific function

JavaScript and Methods

- JavaScript is not fully object-oriented
- However, standard JavaScript does support creation of object-like entity
- This “object” is able to reference associated properties and methods of object
- It uses conventional Object Oriented (OO) “dot” syntax
- Some fuller OO implementations of JS exist

JavaScript Object Methods

- It is possible to create objects and call methods on those objects
  - var student80765432 = new person("Fred", "Smith", 28, "long", "red");
  - student80765432.getPersonDetails();
  - student80765432.setAge(48);
- In C# (fully OO) methods may be called directly by objectName.methodName()
- In JavaScript the method setAge() is in fact a variable reference to another function
  - student80765432.setAge(48);
  - this.setAge = changeAge
- Function is declared separately
  - function changeAge(age) { this.age=age; }



```
Fred
Smith
28
long
red

Fred
Smith
48
long
red
```

The Document Object Model (DOM)

- JavaScript has a hierarchical object map of HTML document elements
- At the "root" of the hierarchy is the ***window*** object (*not* the document)
- Thus the full version of
  - document.write("hello world") is
  - window.document.write("hello world")
- McGrath's representation of the DOM (figure on the left) is selected as it illustrates the important "path" to "form" elements

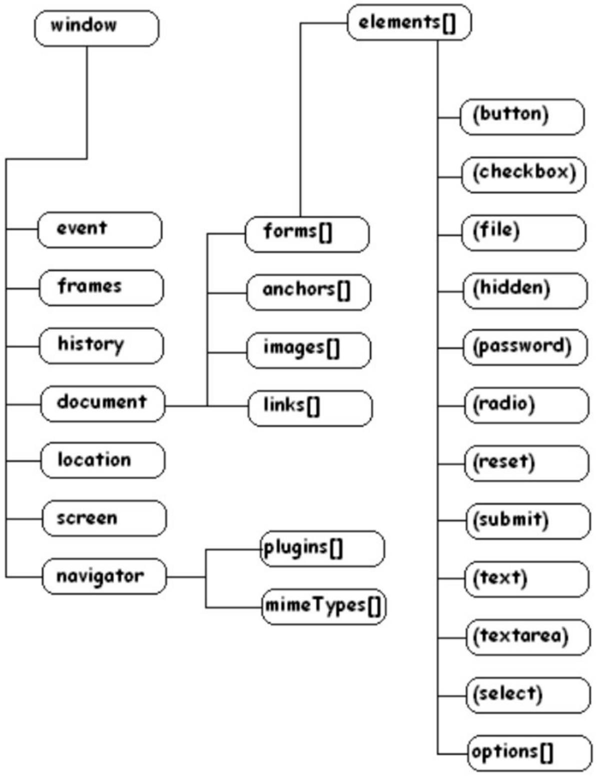
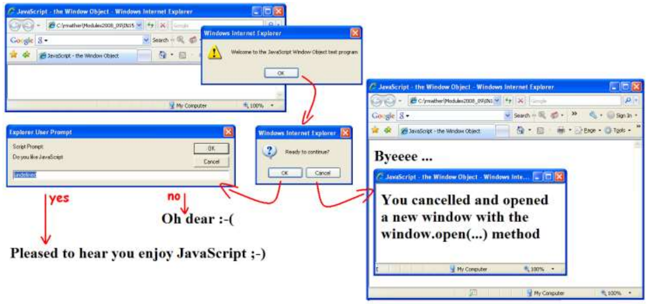


Figure is adapted from: McGrath M (2006) ***JavaScript in easy steps.*** Computer Step.

The *window* object

- We have already used ***alert()***, ***confirm()*** and ***prompt()*** popups – all are methods of the window object.
- Can also window.open(…) a new window and set some properties simultaneously
  - var win = window.open("openNewWindow_v1.html", "testJS", "width=400, height=350, left=450, top=250, status=yes, resizable=yes");
  - Parameters are: [1] the URL for the new HTML document; [2] a name, can be use to target output with <a> or <form> tags; [3] a string controlling widow display properties
- Click image to see the above and the window ***status*** bar visible property (when it works in IE!)



Welcome to the JavaScript Window Object test program

Script Prompt:
Do you like JavaScript

Ready to continue?

Pleased to hear you enjoy JavaScript ;-)

Oh dear :-(

Byeeee ...

You cancelled and opened a new window with the window.open(...) method

JavaScript - the Window Object - Windows Internet Explorer
Windows Internet Explorer
OK
Explorer User Prompt
OK
Cancel
undefined
Windows Internet Explorer
OK
Cancel
yes
no
My Computer
100%

The ***window*** object - continued

- As well as ***window.open()*** can ***explicity window.close()***
- ***window*** has many properties and methods (see Table 6-2, p135 in Mosley, 2006)
- JavaScript also generates a ***screen*** object that can be used to return the dimensions and colour depth of the users display (thus modify HTML pages on the fly accordingly)

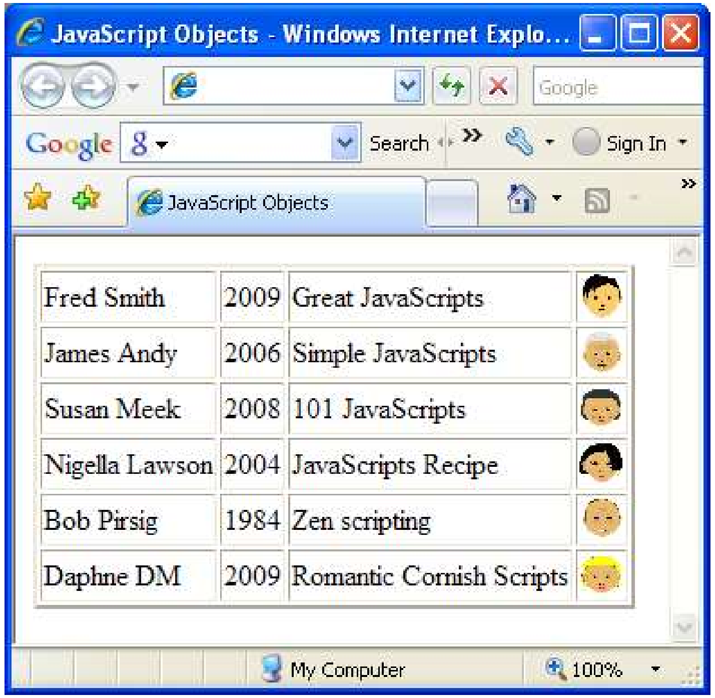
JavaScript summary

- Introduction to JavaScript objects and the DOM
- JavaScript’s internal objects and user-defined objects
- The construction of objects
- The declaration and use of object properties and methods

- NEXT WEEK …
- the DOM really comes into use with FORMS so …
- … we will finish the DOM in the context of FORMS and the VALIDATION of forms (Moseley – 2006 - s6.3.1 and s6.4 [pp136-145])

Week 8 - Practical

- Modify the person constructor in objectMethods_v1.html to create book objects with properties for author, date of publication, title and a picture of the author.
- Modify the showDetails() method to output all properties of one book in one HTML table row (i.e. using <tr> and <td> tags)
- Modify the script in the body element to declare and initialise an array of book objects and populate
- the elements of the array by
- constructing six book objects
- Use a for loop to iterate through the array and generate output of the type shown in the figure.
- ***OPTIONAL*** EXTRA
  Improve the lottery numbers application
  (slide 3 - mathObject_v1.html) by:
  [1] sorting selection output in ascending order; and
  [2] ensuring all numbers are unique?



Fred Smith	2009	Great JavaScripts	
James Andy	2006	Simple JavaScripts	
Susan Meek	2008	101 JavaScripts	
Nigella Lawson	2004	JavaScripts Recipe	
Bob Pirsig	1984	Zen scripting	
Daphne DM	2009	Romantic Cornish Scripts	