

Introduction to JavaScript

Prof. Ing. Andrea Omicini
II Facoltà di Ingegneria, Cesena
Alma Mater Studiorum, Università di Bologna
andrea.omicini@unibo.it

Documents and computation

- HTML
 - Language for the description of documents
 - Information-oriented
 - Document mobility
 - Distributed information
- How to distribute computation using the Web?
 - Associating mobile code to HTML pages
 - Applet Java
 - JavaScript

2

JavaScript vs. Java Applet

- Specialisation on the "client as browser" model
- Dynamics
- "Lightness"
- Regular Expressions agile management
 - Perl-like
- Weakly typed
 - easy prototyping
- Inheritance and objects
 - prototype vs. class
- ...

3

Myths

- JavaScript is similar to Java
 - A little...
- JavaScript is simple
 - It is easily usable without training
- JavaScript runs on every browser
 - Yes, of course, thankyouverymuch
 - but it can have specific quirks on specific browsers
 - Versions, IE vs Mozilla (Netscape) vs Opera vs ...
- ECMA
 - <http://www.ecma-international.org/>

4

Standard

- ECMA 262
 - ISO 16262
 - ECMAScript
 - JavaScript, Jscript

<http://www.ecma-international.org/publications/standards/ECMA-262.HTM>
<http://www.ecma-international.org/publications/files/ecma-st/ECMA-262.pdf>
- ECMA 357
 - E4X
 - ECMAScript for XML

<http://www.ecma-international.org/publications/standards/ECMA-357.HTM>
<http://www.ecma-international.org/publications/files/ecma-st/ECMA-357.pdf>

5

JavaScript

- Object-oriented / Functional language
 - Model
 - Syntactic details
- Client side
 - Browser integration
- Server side
 - We are not interested
- Embedded
 - I have not heard about that in a long time...

6

Example - XHTML

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head>
  <meta http-equiv="content-type" content="text/html; charset=utf-8" />
  <title>_</title>
  <link href="style.css" rel="stylesheet" type="text/css" media="screen" />
  <script type="text/javascript" src="command.js"></script>
</head>
<body class="papers">
-
<form action="" method="get">
-
<input type="button" value="BibTeX" class="bibBtn" onClick="showBib('volume');">
-
```

7

Example - JavaScript

```
absURL = "abs/";
bibURL = "bib/";
pdfURL = "pdf/";

function showAbs(key) {
  abstractWin=window.open(absURL+key+".html", "abstractWindow",
    "resizable=yes, dependent=yes, height=150, width=600, location=no, menubar=no,
    scrollbars=yes, status=no, toolbar=no");
  abstractWin.focus();
}

function showBib(key) {
  bibtexWin=window.open(bibURL+key+".html", "bibtexWindow",
    "resizable=yes, dependent=yes, height=300, width=600, location=no, menubar=no,
    scrollbars=yes, status=no, toolbar=no");
  bibtexWin.focus();
}

function showPDF(key) {
  top.location.href=pdfURL+key+".pdf";
}
-
```

8

What does JavaScript do?

- 1 Document content and presentation control
 - 2 The document object
 - 3 DOM
- 4 Browser control
 - 5 The window object
- 6 Form management
 - 7 The Form, Button, ... objects
- 8 User interaction
 - 9 Events management
 - 10 Interaction state management
 - 11 Cookies

9

Structure of the language

- 1 Case sensitive
 - 2 It is a problem using HTML
- 3 Separators
 - 4 Spaces, line breaks, tabs, ...
- 5 Semicolon
 - 6 Optional, but please use it
- 7 Comments
 - 8 Similar to C, C++ e Java
 - 9 Use // for single line and /* ... */ for multiline
- 10 Keywords

10

Data types

- 1 Primitive types
 - 2 Number, string and boolean
- 3 Objects
 - 4 General e special
 - 5 window, document, Data, RegExp, ...
- 6 Array
- 7 Functions
- 8 E4X adds XML-like data types

11

Numbers

- 1 Integer and real numbers as IEEE 8 byte
 - 2 Only double-precision numbers
- 3 The Math object
 - 4 Library of mathematical functions
- 5 Special values
 - 6 Infinity
 - 7 NaN
 - 8 ...

12

Strings

- No char type
- Quotes and double quotes
 - they are equal
 - pay attention with (X)HTML
- Concatenation
 - and many other "classic" operators
 - Wrapper string
 - Virtual "library", à la Java (static functions)

13

Boolean

- false and true
 - As strings
- Automatically converted in 0 and 1
 - Numbers
 - Whenever needed...

14

Primitive types and references

- Assignment
 - Between non-primitive types
 - References are shared
- Example

```
var a = [1,2,3];
var b = a;
a[0] = 99;
alert(b);
```

 - what does that do?
- Try it! (IE, Mozilla, Opera, Safari/Konqueror)

```
javascript: var a = [1,2,3]; var b = a; a[0] = 99; alert(b);
```

 - what is the output?

15

Variables and scope

- Keyword var
 - Used or not...
- Scopes
 - Global
 - Global object
 - Local
 - Execution context
 - No blocks
- Web
 - documents and windows are new context in addition to "classic" scopes

16

Expressions and operators

- We won't present them
 - Similar to C, C++, Java, more or less...
 - Please help yourself...
- There's all sort of them
- Note
 - `typeof`
 - a kind of "reflexive" operator

17

Control instructions

- Selection
 - if, if/else, else if
 - switch
- Iteration
 - while, do/while, for, for/in
- Function
 - function, return
- ...

18

Functions

- First class objects
 - Parameters
 - Lambda expression, closures

- Examples

```
function square(x) {return x*x;}  
var square = new Function("x", "return x*x;");  
var square = function(x) {return x*x};
```

- Function objects and properties

- The "call" object
 - arguments, caller
- length and arity
- apply and call

19

Objects

- Collections of properties
 - with names
- The new operator

```
var paper = new Object();
```
- Definition of / access to properties

```
paper.title = "JavaScript -- Ohboy!!!";
```
- Enumeration

```
for/in
```
- Methods
 - Properties like any other
- Prototypes
 - Not (only) classes and inheritance
 - In the 3rd standard, class and prototype properties...

20

Array

- As objects...
 - `var arr = new Array(1,2,3,4,5);`
- Classic access
 - `var four = arr[3];`
 - `var arr = [[2,3],[true,false],["boh", 'mah']];`
- Fragmented and dynamic
 - you can do everything you want...
- Wrapper Array

21

Regular Expressions

- Excellent to manage text
 - User input
- The `RegExp` object
- A lot of relevant details...
 - OK, let's move on!

22

Browser integration

- The window object
 - Window as a global execution context
 - `var foo` and `window.foo` are the same
- Client-side object hierarchy
 - The window object contains
 - `document`, `location`, `frames[]`, `forms[]`, ...
- Event model
 - Event managers associated to (X)HTML tags

23

The SCRIPT tag

```
...  
<html>  
<head>  
...  
<script type="text/javascript" language="JavaScript">  
<!-- hide to very old browsers  
    javascript code  
-->  
</script>  
</head>  
...
```

24

Windows management

- 1 You can control almost everything...
 - 2 but you need to study a little
 - 3 so it is better to start from existing examples...
- 4 A window objects hierarchy
 - 5 screen, navigator, document, ...

```
function showBib(key) {
  bibtexWin=window.open(bibURL+key+".html","bibtexWindow",
    "resizable=yes,dependent=yes,height=300,width=600,location=no,menubar=no,
    scrollbars=yes,status=no,toolbar=no");
  bibtexWin.focus();
}
function showPDF(key) {
  top.location.href=pdfURL+key+".pdf";
}
```

25

DOM

- 1 Standardize the Document Object Model...
 - 2 It has been done in theory, but in practice there are still some problems
 - 3 especially using frames
 - 4 To dynamically generate objects...
- document.write(), writeln(), open(), close()

26

Events

- 1 Event managers
 - 2 onChange, onClick, onMouseDown, onSubmit, ...
- 3 Problem
 - 4 to define a set of common events between IE and the other browsers...
 - 5 it has been tried...
- 6 Managers as HTML attributes

```
...
<form action="" method="get">
...
<input type="button" value="BibTeX" class="bibBtn"
onclick="showBib('volume');">
...
```

27

HTML and Forms

- 1 Every (X)HTML element can have an identifier
 - 2 The id attribute (once called name)
- 3 The form object
 - 4 Modules as elements of document.forms[]
 - 5 Input elements as elements of document.forms[].elements[]
 - 6 Associative access using the name/id name
- 7 The onSubmit() and reset() methods
 - 8 If onSubmit() returns false, data are not sent
 - 9 A crystal-clear example of "distributed computation"...

28

Security

- 1 Implicit
 - 2 No access to the local file system
 - 3 No direct network functions
- 4 Explicit
 - 5 Restricted or privilege based functionality
 - 6 "From the same origin" rule
 - 7 Signed script

29

JavaScript in a few hours?

- 1 Tutorial on the Internet
 - 2 Course website
 - 3 or <http://www.google.it>, search: JavaScript Tutorial
- 4 Example
 - 5 <http://www.pageresource.com/jscript/>
 - 6 tutorial page
 - 7 <http://academ.hvcc.edu/~kantopet/old/javascript/>
- 8 Books
 - 9 "JavaScript: The Definitive Guide" (David Flanagan, O'Reilly/Apogeo)
 - 10 or anything you like...

30