AN INTRODUCTION TO WEB PROGRAMMING

Dr. Hossein Hakimzadeh
Department of Computer and Information Sciences
Indiana University
South Bend, IN

Java Script A

CONTENTS

- Fundamental programming constructs:
  - Variable,
  - Arithmetic operators,
  - Assignment operator,
  - Input and output,
  - Arrays,
  - Loops,
  - Conditions,
  - Modules (Functions),
  - Structures, classes and objects,
  - Events and event-handling
  - Files and Databases

JAVA VS. JAVA-SCRIPT

- **Java**
  - Java is an Object Oriented Programming (OOP) language created by James Gosling of Sun Microsystems.
  - Java is a stand alone language and runs on a virtual machine.
  - Java is compiled in a byte-code (an intermediate machine language) and produces a stand-alone executable.
  - Object-oriented: (Encapsulation, inheritance, polymorphism)
  - Strongly typed language.

- **Java-Script**
  - JavaScript is a scripting language that was created by the fine people at Netscape and was originally known as LiveScript.
  - JavaScript must be part of a HTML document and runs within a browser.
  - JavaScript is interpreted line-by-line by the browser.
  - Object-based: Code uses built-in, extensible objects, but no classes or inheritance.
  - Dynamically typed language

Lean More:
http://en.wikipedia.org/wiki/JavaScript
http://sislands.com/coin70/week1/javajs.htm

WHAT IS JAVA SCRIPT?

- Java script is a scripting language which is typically used to enhance the functionality and appearance of web pages.
- Java script is the de facto standard for client side programming for web based applications.
- Java scripts are executed by the browser, however some browsers (specifically Microsoft IE) disable this capability. Therefore, if you use IE, you need to enable the execution of java script in your browser.

JAVA SCRIPT

- The java script example creates a simple function for displaying the date, and connects it to the event handler of a button.

Example:

```
<!DOCTYPE html>
<html>
<head>
<script>
function displayDate()
{
document.getElementById("placeholder").innerHTML=Date();
}
</script>
</head>
<body>

<h1>My First JavaScript</h1>
<p id="placeholder">the date will appear here...</p>

<button type="button" onclick="displayDate()">Display
    Date</button>

</body>
</html>
```

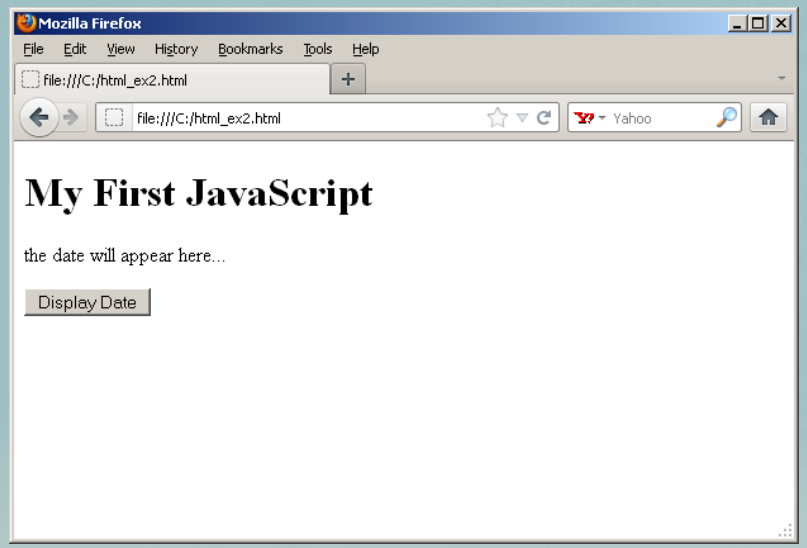
Java Script

Called when the button is clicked

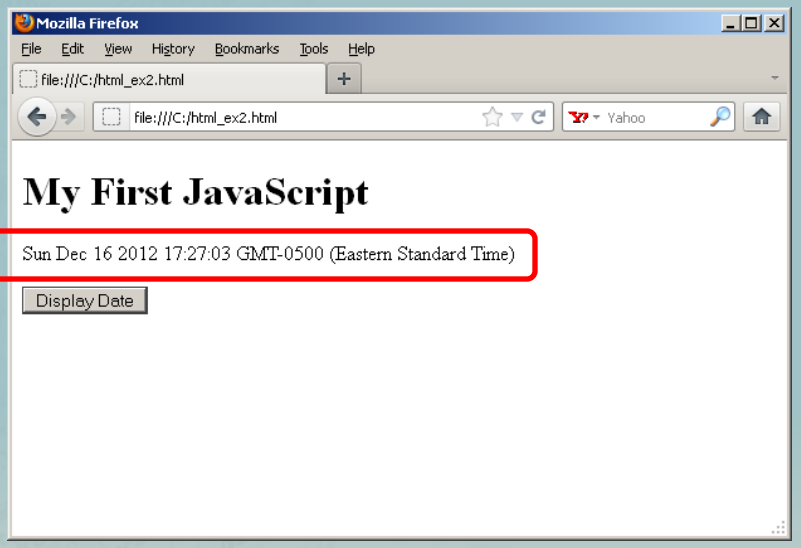
Event Handler

OUTPUT

- Before Click



- After Click



JAVASCRIPT AND ITS RELATION WITH DOM VS. BOM

- **Document Object Model (DOM)**

- The Document Object Model (DOM) is an application programming interface (API) for HTML as well as XML.

- The DOM organizes the entire web page as a document composed of a hierarchy of nodes like a tree structure and using the DOM API, nodes can be removed, added, and replaced.

- **DOM allows the developer to manipulate the document.**

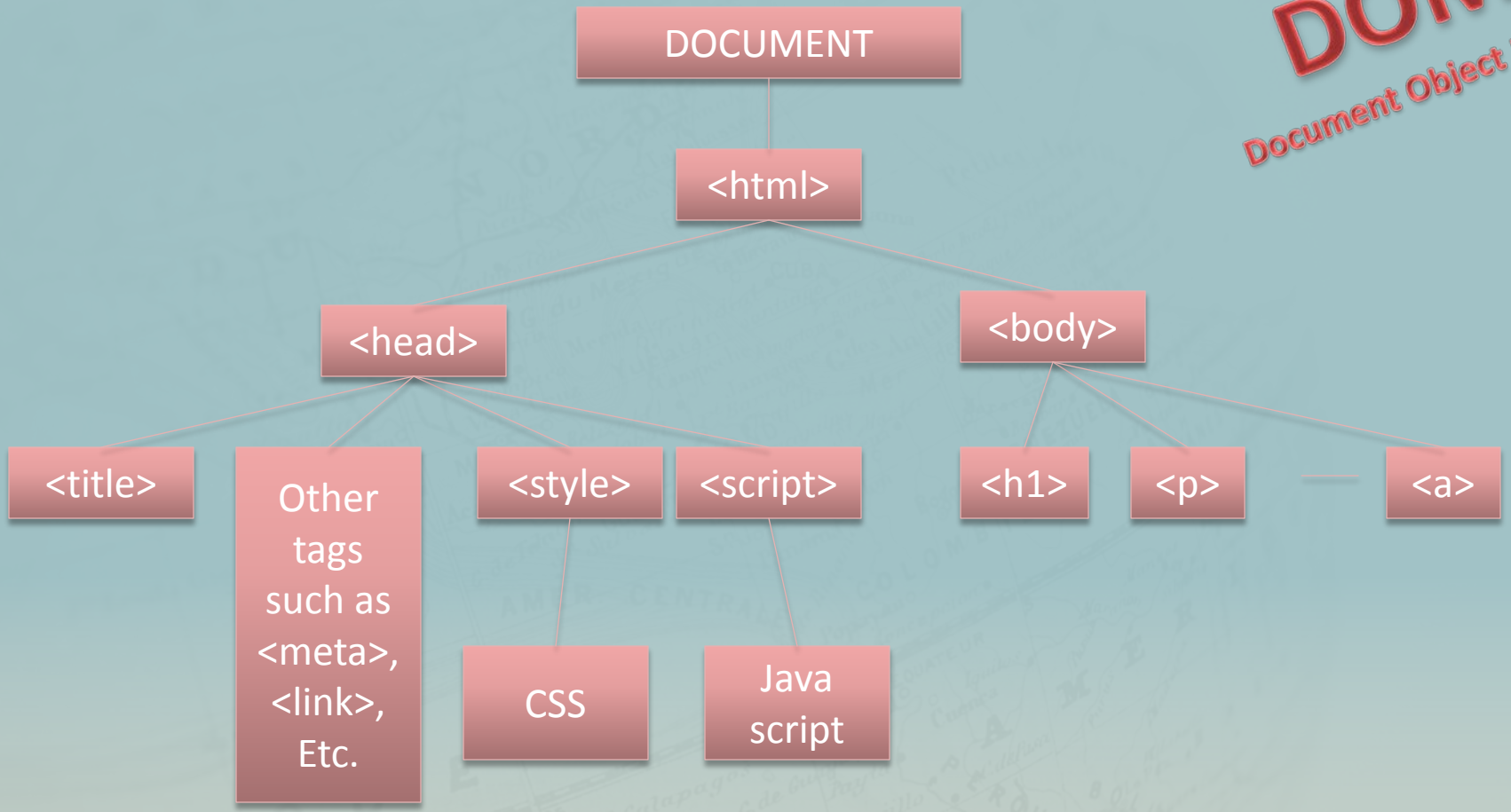
Browser Object Model (BOM)

- Browsers feature a Browser Object Model (BOM) that allows access and manipulation of the browser window. For example (browser history, location, navigator, and screen)

- Because no standards exist for the BOM, each browser has its own implementation.

- **BOM allows the developer to manipulate the browser window.**

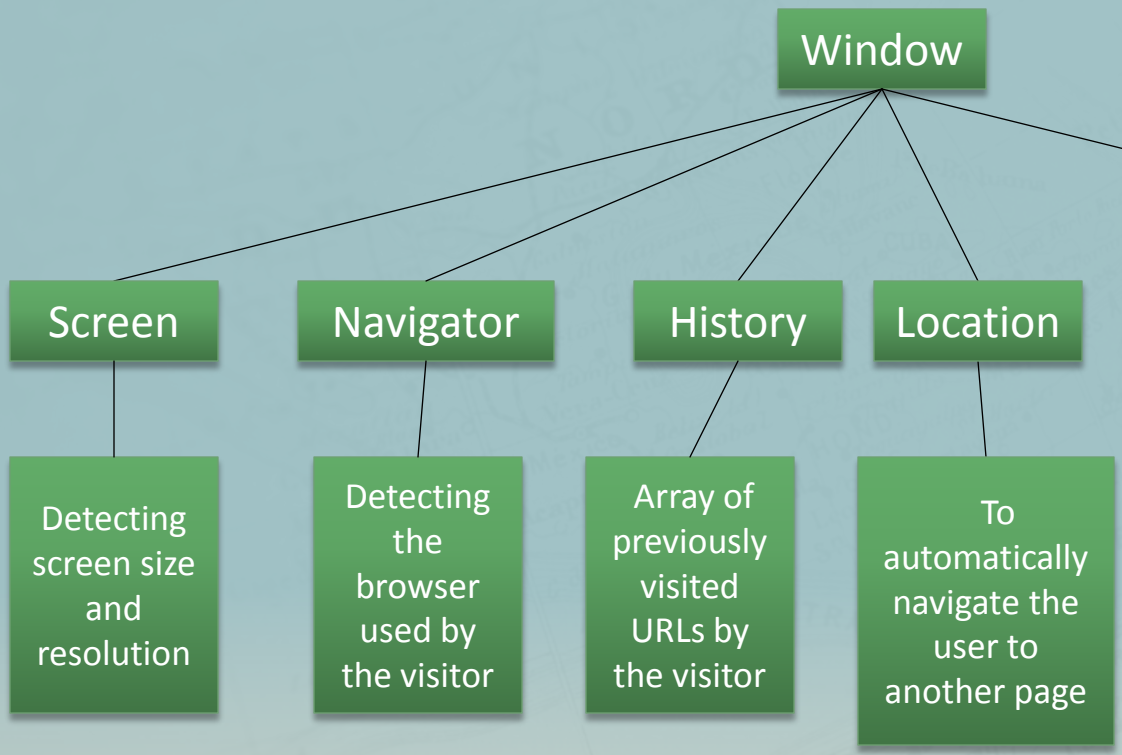
THE STRUCTURE OF AN HTML DOCUMENT

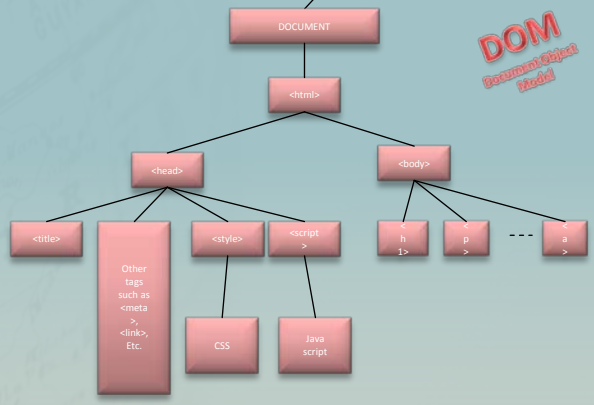


DOM

Document Object Model

THE STRUCTURE OF BROWSER OBJECT





BOM
Browser Object Model

DOM
Document Object Model

http://www.javascriptkit.com/jsref/window.shtml

AN INTRODUCTION TO WEB PROGRAMMING

Dr. Hossein Hakimzadeh
Department of Computer and Information Sciences
Indiana University
South Bend, IN

Java Script B

JAVA-SCRIPT SYNTAX

- Both Java and JavaScript syntax are based on the C and C++ language so it is easy to learn the basics.

VARIABLES

- Variables in JavaScript are dynamically typed.
- Their type is determined only after they are assigned a value.
- Example:

```
var myName;                 // No data type yet.  (undefined)
var myNumber;

myName = "Cyrus";           // variable type will become string
myNumber = 5;               // variable type will become int

myNumber = 5.6;             // change the type int to float
myName = 4;                 // change the type from string to int.
```

VARIABLES

C:\wamp\www\A340\02_HTML_HEAD_Script_5.html - Notepad++

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

02_HTML_HEAD_Script_5.html | 02_HTML_HEAD_Script_5a.html

```
1  <!DOCTYPE html>
2  <!-- 02_HTML_HEAD_Script_5.html
3  variables
4
5  -->
6  <html>
7  <head>
8  <script type="text/javascript">
9
10     var myName;
11     var myNumber;
12     document.writeln("myName   = " + myName   +"<br />");
13     document.writeln("myNumber = " + myNumber +"<hr />");
14
15     myName   = "Cyrus";
16     myNumber = 5;
17     document.writeln("myName   = " + myName   +"<br />");
18     document.writeln("myNumber = " + myNumber +"<hr />");
19
20     myName   = 4;
21     myNumber = 5.6;
22     document.writeln("myName   = " + myName   +"<br />");
23     document.writeln("myNumber = " + myNumber +"<hr />");
24
25 </script>
26 </head>
27 <body>
28
29 </body>
30 </html>
31
```

length : 618 lines : 31 | Ln : 21 Col : 20 Sel : 0 | Dos\Windows | ANSI | INS

Mozilla Firefox

File Edit View History Bookmarks Tools Help

file:///C:/wamp/www/...L_HEAD_Script_5.html

myName = undefined
myNumber = undefined

---

myName = Cyrus
myNumber = 5

---

myName = 4
myNumber = 5.6

ARITHMETIC OPERATORS

- Same as C/C++ operators

  +, -, *, /, %, ++, --

OTHER OPERATORS

- Bitwise

  http://www.javascriptkit.com/jsref/bitwise_operators.shtml

  &, |, ^, ~, >>, <<

- Comparison

  ==, !=, < , >, <=, >=

  http://www.javascriptkit.com/jsref/comparison_operators.shtml

- Logical

  &&, ||, !

- Other:

  typeof, new, delete

  http://www.javascriptkit.com/jsref/other_operators.shtml

TYPEOF

C:\wamp\www\A340\02_HTML_HEAD_Script_5a.html - Notepad++

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

02_HTML_HEAD_Script_5.html | 02_HTML_HEAD_Script_5a.html

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

    var myName;
    var myNumber;
    document.writeln("myName    = " + myName    +"<br />");
    document.writeln("myNumber  = " + myNumber  +"<br />");
    document.writeln("typeof myName =" + typeof(myName) +"<br />");
    document.writeln("typeof myNumber =" + typeof(myNumber) +"<hr />");


    myName    = "Cyrus";
    myNumber = 5;
    document.writeln("myName    = " + myName    +"<br />");
    document.writeln("myNumber  = " + myNumber  +"<br />");
    document.writeln("typeof myName =" + typeof(myName) +"<br />");
    document.writeln("typeof myNumber =" + typeof(myNumber) +"<hr />");


    myName    = 4;
    myNumber = 5.6;
    document.writeln("myName    = " + myName    +"<br />");
    document.writeln("myNumber  = " + myNumber  +"<br />");
    document.writeln("typeof myName =" + typeof(myName) +"<br />");
    document.writeln("typeof myNumber =" + typeof(myNumber) +"<hr />");

</script>
</head>
<body>

</body>
</html>
```

length : 1041 lines : 39 | Ln : 31 Col : 73 Sel : 0 | Dos\Windows | ANSI | INS

Mozilla Firefox

File Edit View History Bookmarks Tools Help

file:///C:/wamp/www/..._HEAD_Script_5a.html

myName = undefined
myNumber = undefined
typeof myName =undefined
typeof myNumber =undefined

---

myName = Cyrus
myNumber = 5
typeof myName =string
typeof myNumber =number

---

myName = 4
myNumber = 5.6
typeof myName =number
typeof myNumber =number

OUTPUT

- Output methods in JavaScript apply to the DOM's DOCUMENT object.

- Example:

  document.**write**("hello");

  document.**writeln**("hello");

Remember however, document is part of a window (or BOM object)

```
myWin=window.open("");                  //open blank window and write to it
myWin.document.open();                  //open document stream
myWin.document.write("Hello");
myWin.document.close();
```

INPUT

- Input methods in JavaScript apply to the BOM's WINDOW object.

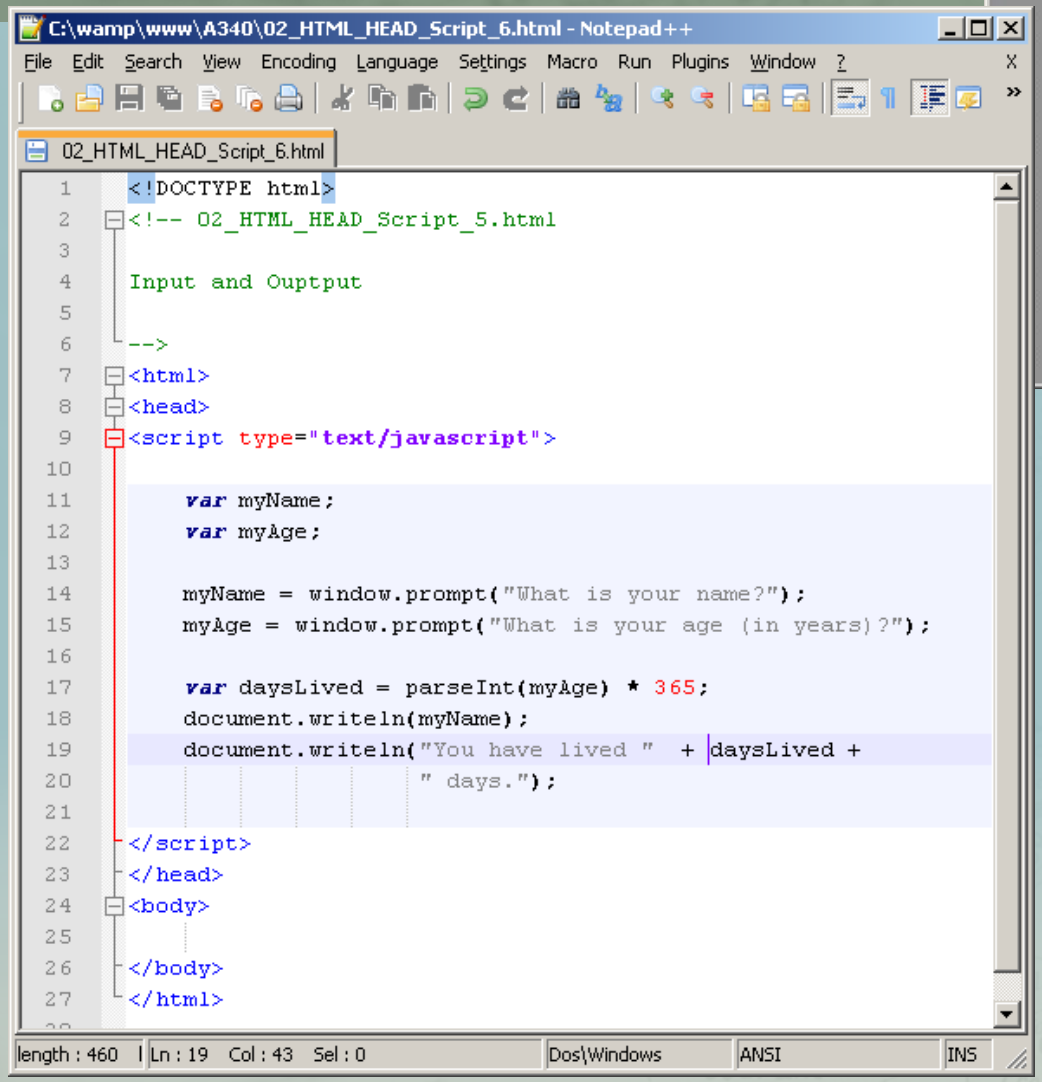
- Example:

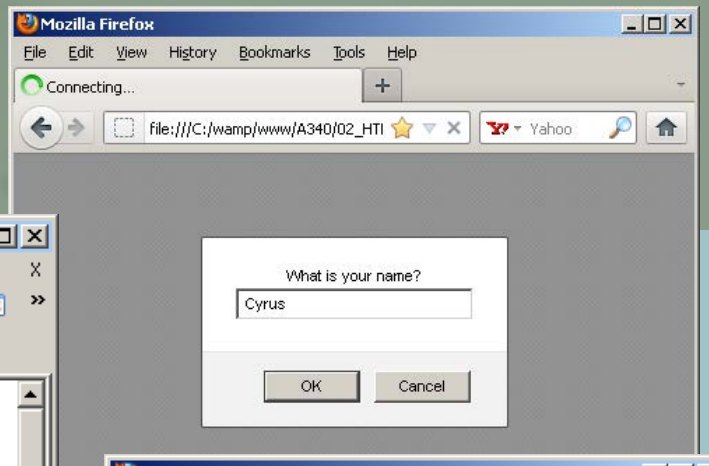
```
window.prompt(msg, [input]);

var myName;
myName = window.prompt("please enter your name")
```

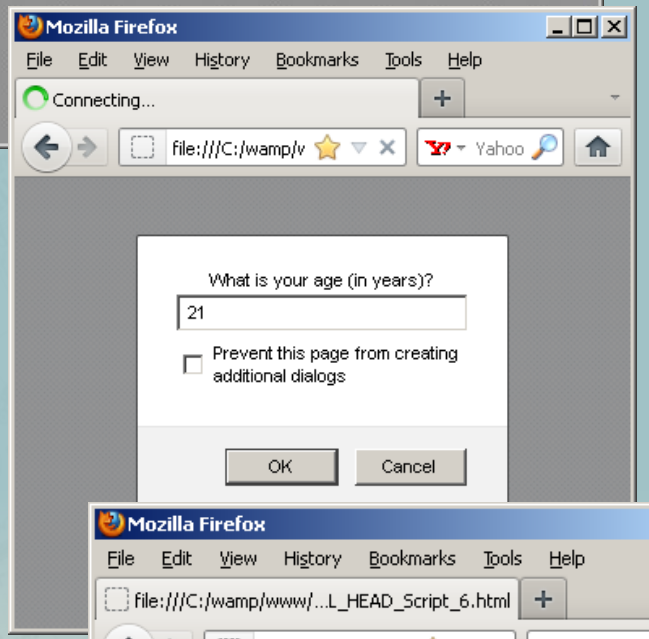
- Displays a Prompt dialog box with a message. Optional "input" argument allows you to specify the default input (response) that gets entered into the dialog box.

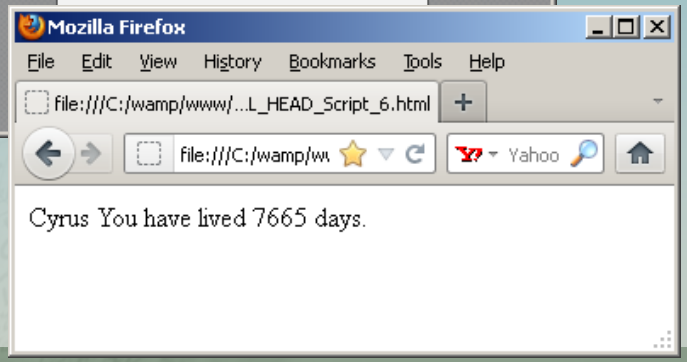
- Prompt will return the string the user has entered.

INPUT


C:\wamp\www\A340\02_HTML_HEAD_Script_6.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
02_HTML_HEAD_Script_6.html
1 <!DOCTYPE html>
2 <!-- 02_HTML_HEAD_Script_5.html
3
4 Input and Ouptput
5
6 -->
7 <html>
8 <head>
9 <script type="text/javascript">
10
11 var myName;
12 var myAge;
13
14 myName = window.prompt("What is your name?");
15 myAge = window.prompt("What is your age (in years)?");
16
17 var daysLived = parseInt(myAge) * 365;
18 document.writeln(myName);
19 document.writeln("You have lived " + daysLived +
20 " days.");
21
22 </script>
23 </head>
24 <body>
25
26 </body>
27 </html>
length : 460 Ln : 19 Col : 43 Sel : 0 Dos\Windows ANSI INS


Mozilla Firefox
File Edit View History Bookmarks Tools Help
Connecting...
file:///C:/wamp/www/A340/02_HTI
Yahoo
What is your name?
Cyrus
OK
Cancel


Mozilla Firefox
File Edit View History Bookmarks Tools Help
Connecting...
file:///C:/wamp/v
Yahoo
What is your age (in years)?
21
Prevent this page from creating additional dialogs
OK
Cancel


Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///C:/wamp/www/...L_HEAD_Script_6.html
file:///C:/wamp/w
Yahoo
Cyrus You have lived 7665 days.

AN INTRODUCTION TO WEB PROGRAMMING

Dr. Hossein Hakimzadeh
Department of Computer and Information Sciences
Indiana University
South Bend, IN

Java
Script
C

ARRAY

- Can be allocated statically or dynamically.

Dynamic arrays:

```
var mySchedule = new Array([size]);
mySchedule[0] = “CSCI-A 340”;
mySchedule[1] = “MATH-M 215”;
```

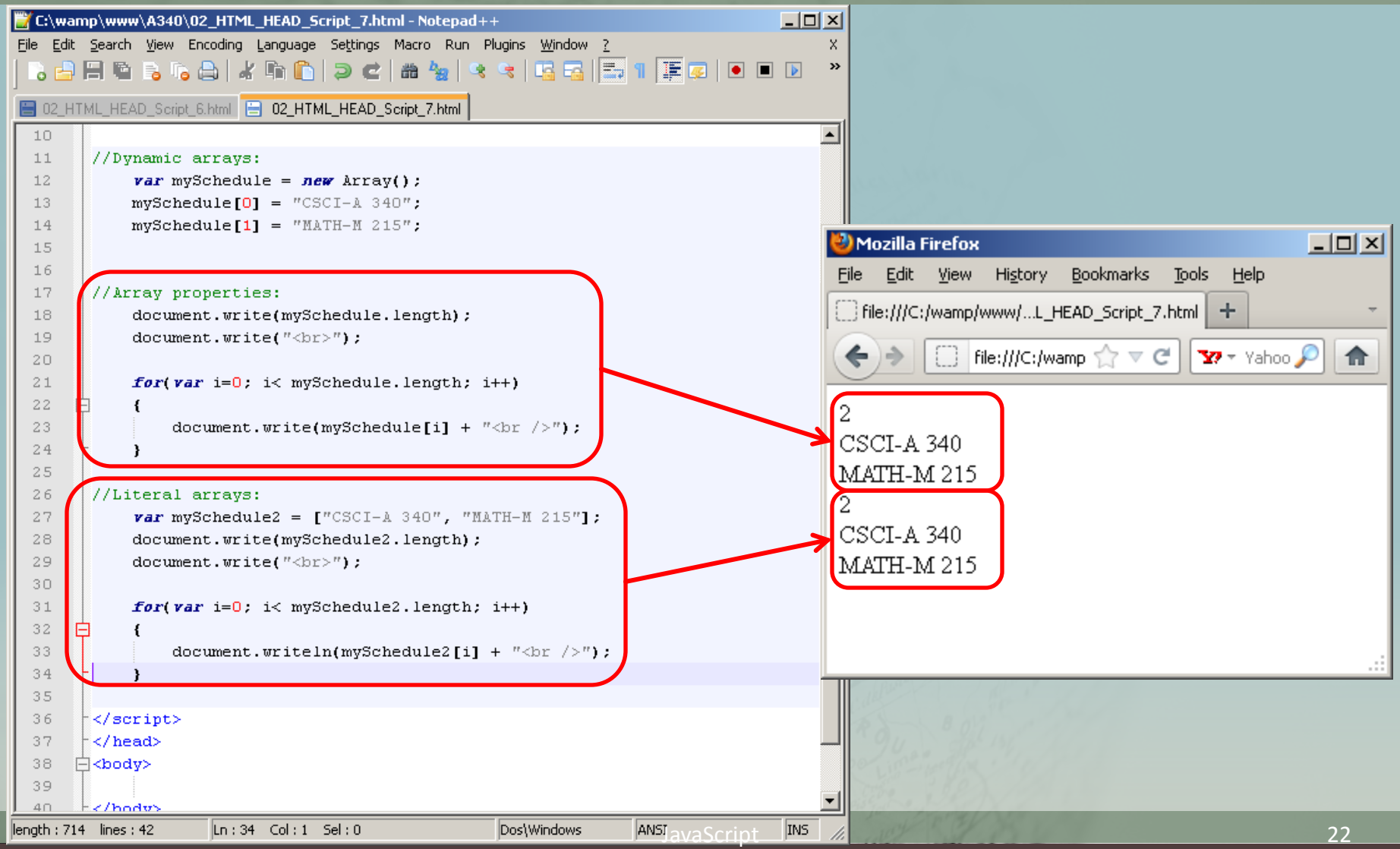
Literal arrays:

```
var mySchedule = [“CSCI-A 340”, “MATH-M 215”];
```

Array properties:

```
mySchedule.lenght
```

ARRAY


C:\wamp\www\A340\02_HTML_HEAD_Script_7.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
02_HTML_HEAD_Script_6.html
02_HTML_HEAD_Script_7.html
10
11 //Dynamic arrays:
12 var mySchedule = new Array();
13 mySchedule[0] = "CSCI-A 340";
14 mySchedule[1] = "MATH-M 215";
15
16
17 //Array properties:
18 document.write(mySchedule.length);
19 document.write("<br>");
20
21 for(var i=0; i< mySchedule.length; i++)
22 {
23 document.write(mySchedule[i] + "<br />");
24 }
25
26 //Literal arrays:
27 var mySchedule2 = ["CSCI-A 340", "MATH-M 215"];
28 document.write(mySchedule2.length);
29 document.write("<br>");
30
31 for(var i=0; i< mySchedule2.length; i++)
32 {
33 document.writeln(mySchedule2[i] + "<br />");
34 }
35
36 </script>
37 </head>
38 <body>
39
40 </body>
length : 714 lines : 42
Ln : 34 Col : 1 Sel : 0
Dos\Windows
ANSI
INS
Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///C:/wamp/www/...L_HEAD_Script_7.html
file:///C:/wamp
Yahoo
2
CSCI-A 340
MATH-M 215
2
CSCI-A 340
MATH-M 215

OPERATION ON ARRAYS

- **Array.sort([SortFunction])**

- By default the function sort(), sorts the arrays alphabetically and in ascending order.

- **//Sort Alphabetically and ascending:**
  var myArray=["Bob","Bully","Amy"];
  myArray.sort(); //Array now becomes ["Amy", "Bob", "Bully"];

  //Sort Alphabetically and descending:
  var myarray=["Bob","Bully","Amy"];
  myArray.sort();
  myArray.reverse(); //Array now becomes ["Bully", "Bob", "Amy"]

- To sort the array numerically, need to compare the relationship between "a" to "b", with a return value of <0 indicating to sort ascending, and >0 to sort descending

- **//Sort numerically and ascending:**
  var myArray=[25, 8, 7, 41];
  myArray.sort(function(a,b){return a - b}); //Array now becomes [7, 8, 25, 41]

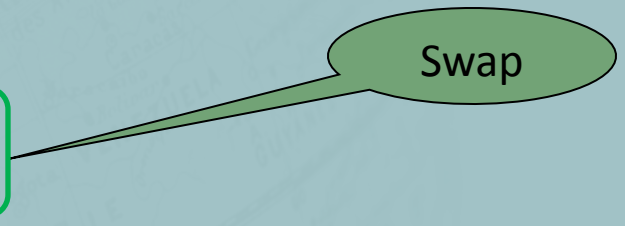
  //Sort numerically and descending:
  var myArray=[25, 8, 7, 41];
  myArray.sort(function(a,b){return b - a}); //Array now becomes [41, 25, 8, 7]

http://www.javascriptkit.com/jsref/arrays.shtml#e2

OPERATIONS ON AN ARRAY:

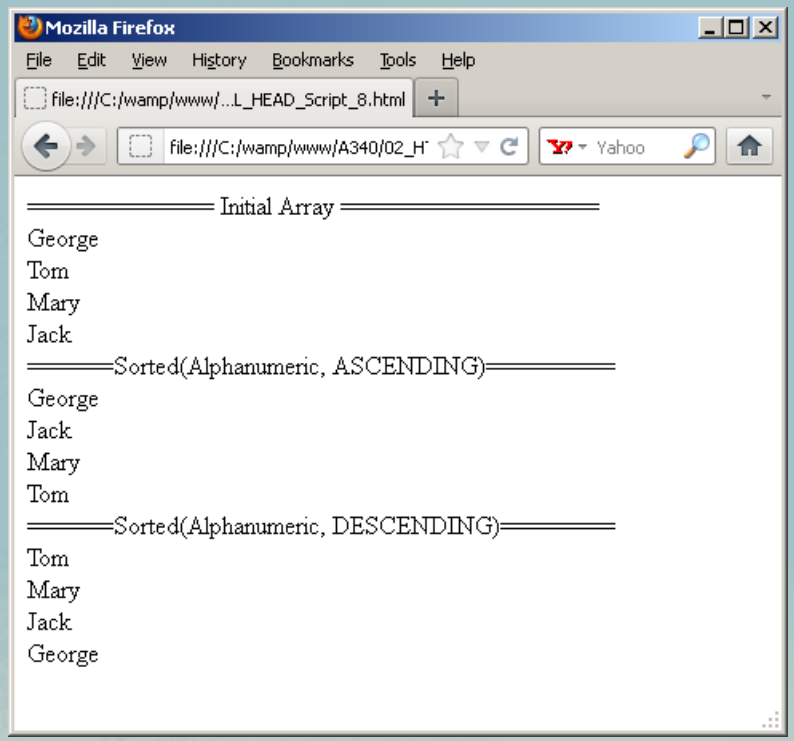
- **Sorting an Array** (Bubble Sort)

```
Private Sub BubbleSort(ByRef TheArray() As Integer)
    Dim Pass, Index, Hold As Integer
    For Pass = 1 To TheArray.GetUpperBound(0)
      For Index = 0 To TheArray.GetUpperBound(0) – 1
        If  (TheArray(Index) > TheArray(Index + 1)  )Then
                Hold = TheArray(Index)
                TheArray(Index) = TheArray(Index + 1)
                TheArray(Index + 1) = Hold
        End If
      Next Index
    Next Pass
End Sub
```



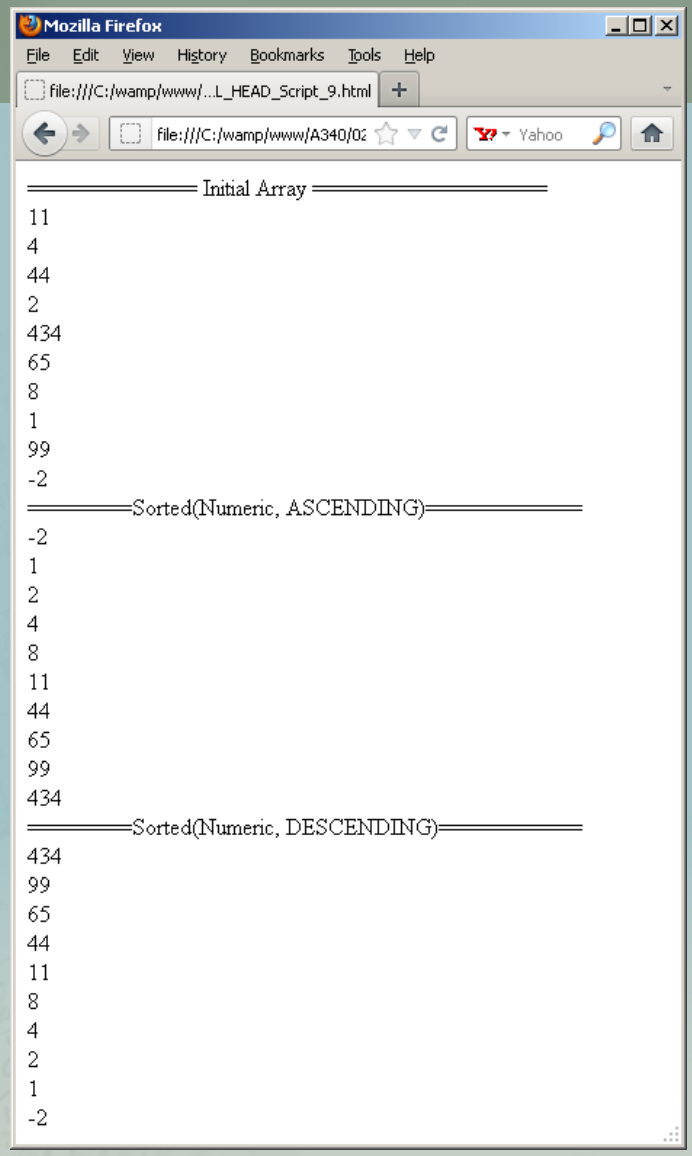
SORTING (ALPHANUMERIC)

```
<!DOCTYPE html>
<!-- 02_HTML_HEAD_Script_8.html
Array Operations
Alphanumeric Sort()
Reverse()
-->
<html>
<head>
<script type="text/javascript">

    document.writeln("============= Initial Array ================== <br>");
    var myArray = ["George", "Tom", "Mary", "Jack"];

    for(var i=0; i< myArray.length; i++)
    {
        document.writeln(myArray[i]);
        document.write("<br>");
    }

    document.writeln("======Sorted(Alphanumeric, ASCENDING)========= <br>");
    myArray.sort();
    for(var i=0; i< myArray.length; i++)
    {
        document.writeln(myArray[i]);
        document.write("<br>");
    }

    document.writeln("======Sorted(Alphanumeric, DESCENDING)======== <br>");
    myArray.reverse();
    for(var i=0; i< myArray.length; i++)
    {
        document.writeln(myArray[i]);
        document.write("<br>");
    }

</script>
</head>
<body>

</body>
</html>
```



SORTING (NUMERIC)

```
<script type="text/javascript">
// ---------------------------------------------------------------------
// Note:
// The following two functions are used by the array.sort() function
// to allow for numeric sorting.
function ASCENDING (a,b) {return a-b}
function DESCENDING (a,b) {return b-a}
// ---------------------------------------------------------------------
    var myArray = [11,4,44,2,434,65,8,1,99,-2];

    document.writeln("============= Initial Array ================== <br>");
    for(var i=0; i< myArray.length; i++)
    {
        document.writeln(myArray[i]);
        document.write("<br>");
    }

    document.writeln("========Sorted(Numeric, ASCENDING)============ <br>");
    myArray.sort(ASCENDING);
    for(var i=0; i< myArray.length; i++)
    {
        document.writeln(myArray[i]);
        document.write("<br>");
    }

    document.writeln("========Sorted(Numeric, DESCENDING)=========== <br>");
    myArray.sort(DESCENDING);
    for(var i=0; i< myArray.length; i++)
    {
        document.writeln(myArray[i]);
        document.write("<br>");
    }

</script>
```



OPERATION ON ARRAYS

Array. Concat()

- Concatenates either a single elements or another array of elements with the existing array, and returns the new array.

```
var fruits=["Apple", "Oranges"];
var meat=["Pork", "Chicken"];

var dinner=fruits.concat(meat);
//creates ["Apple", "Oranges", "Pork", "Chicken"]. fruits and meat arrays not changed.

var snack=fruits.concat("Grapes", ["Cookies", "Milk"]);
//creates ["Apple", "Oranges", "Grapes", "Cookies", "Milk"] fruits array not changed.
```

http://www.javascriptkit.com/jsref/arrays.shtml#e2

CONCATENATION

C:\wamp\www\A340\02_HTML_HEAD_Script_10.html - Notepad++

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

02_HTML_HEAD_Script_10.html

```
<!DOCTYPE html>
<!-- 02_HTML_HEAD_Script_10.html

Array Operations
Concatenation
Concatenates either a single elements or another array of
elements with the existing array, and returns the new array.

-->
<html>
<head>
<script type="text/javascript">

    var male=["Tom", "Jack", "John"];
    var female=["Sue", "Mary", "Anne"];

    female = female.concat("Niki");     //Concat Niki to females
    var myClass =male.concat(female);   //Concatenate the two arrays

//  var myClass =[];
//  myClass = myClass.concat(male);
//  myClass = myClass.concat(female, "Niki");

    document.writeln("<b>Male: </b>" + male + "<br>");
    document.writeln("<b>Female: </b>" +female + "<br>");
    document.writeln("<b>Concat: </b>" + myClass + "<br>");

    myClass.sort();
    document.writeln("<b>Sorted: </b>" + myClass + "<br>");


</script>
</head>
<body>

</body>
</html>
```

length : 882 lines : 38 | Ln : 9 Col : 1 Sel : 0 | Dos\Windows | ANSI | INS

Mozilla Firefox

File Edit View History Bookmarks Tools Help

file:///C:/wamp/www/..._HEAD_Script_10.html

Male: Tom,Jack,John
Female: Sue,Mary,Anne,Niki
Concat: Tom,Jack,John,Sue,Mary,Anne,Niki
Sorted: Anne,Jack,John,Mary,Niki,Sue,Tom

OPERATION ON ARRAYS

Array. indexOf(targetElement)

- Returns the first index in which targetElment (value) is found within an array, or -1 if nothing is found.

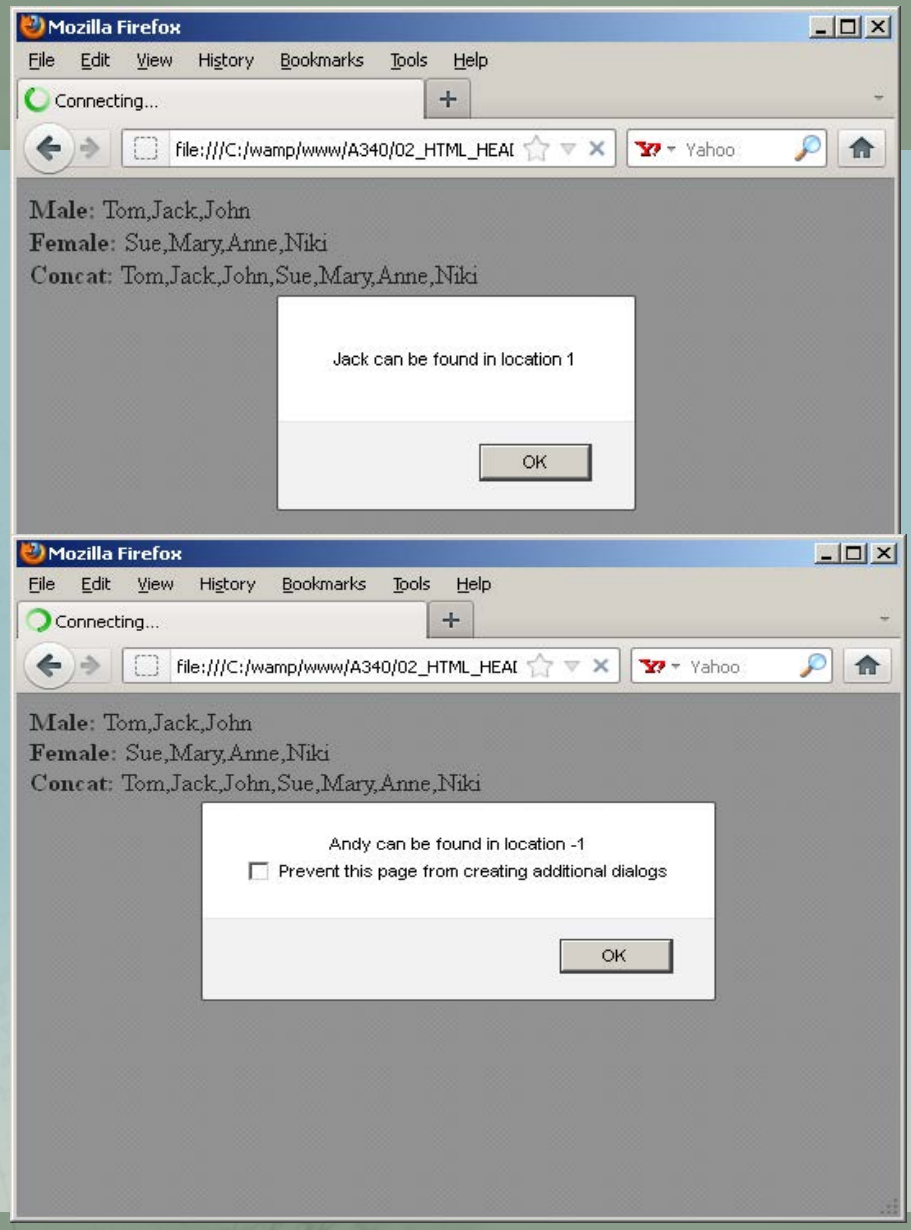
- var fruits=["Apple", "Oranges", "Pork", "Chicken"];

  alert(fruits.indexOf("Pork")); //alerts 2

http://www.javascriptkit.com/jsref/arrays.shtml#e2

INDEX OF

```
<!DOCTYPE html>
<!-- 02_HTML_HEAD_Script_11.html

Array Operations
Array. indexOf(targetElement)

Returns the first index in which targetElment (value) is found within an
array, or -1 if nothing is found.


-->
<html>
<head>
<script type="text/javascript">

    var male=["Tom", "Jack", "John"];
    var female=["Sue", "Mary", "Anne"];

    female = female.concat("Niki");     //Concat Niki to females
    var myClass =male.concat(female);   //Concatenate the two arrays

    document.writeln("<b>Male: </b>" + male + "<br>");
    document.writeln("<b>Female: </b>" +female + "<br>");
    document.writeln("<b>Concat: </b>" + myClass + "<br>");

    alert( "Jack can be found in location " + myClass.indexOf("Jack"));
    alert( "Andy can be found in location " + myClass.indexOf("Andy"));

</script>
```



OPERATION ON ARRAYS

Array. Join([separator])

- Converts each element within the array to a string, and joins them into one large string. Pass in an optional separator as argument to be used to separate each array element. If none is passed, the default comma (,) is used:

Array. Map(mappingfunction())

- Returns a new array based on the return value of testfunction() on each of the array elements. Original array is not changed. Use it to transform the values of all elements within an array using some logic and derive the results as a new array.

Array.push(value);

- Adds the argument values to the end of the array, and modifies the original array with the new additions. Returns the new length of the array.

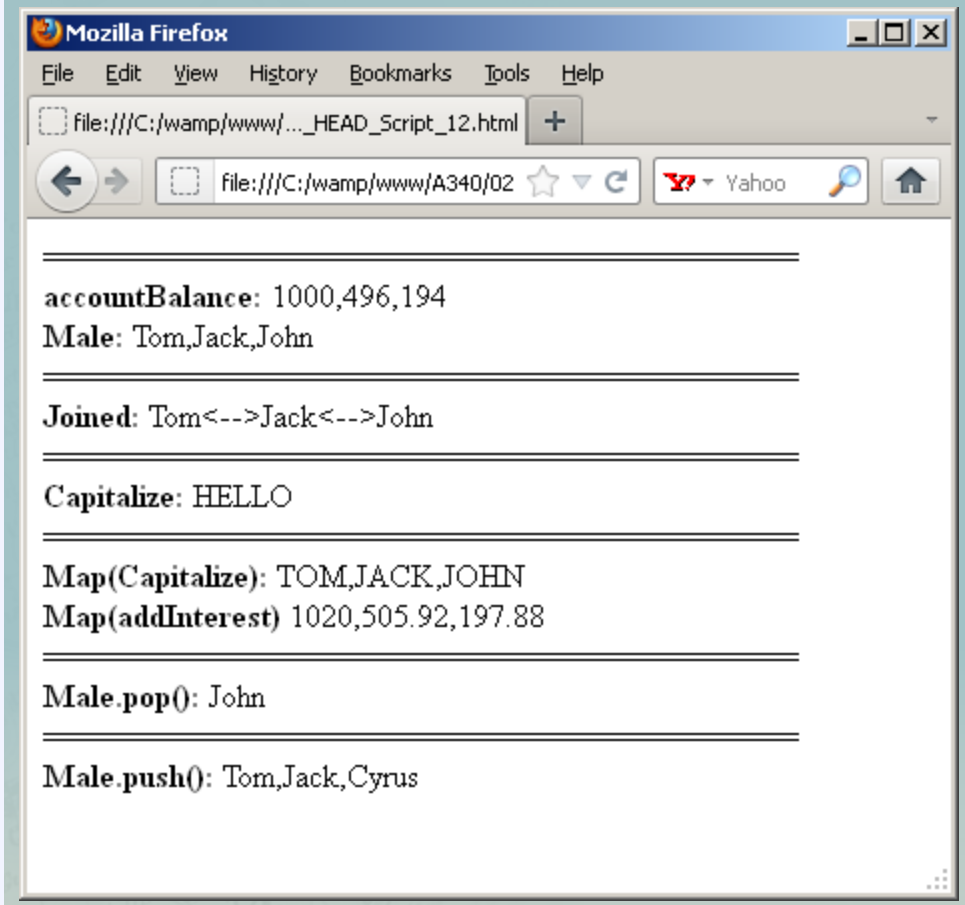
Array.pop();

- Deletes the last element within array and returns the deleted element. Original array is modified.

http://www.javascriptkit.com/jsref/arrays.shtml#e2

OTHER ARRAY OPERATIONS

```
14  <script type="text/javascript">
15      function capitalize(theString)
16      {
17          return (theString.toUpperCase());
18      }
19
20      function addInterest(element) {
21          return (element*1.02);
22      }
23
24      var accountBalance=[1000, 496, 194];
25      var male=["Tom", "Jack", "John"];
26
27      document.writeln("=========================================<br>");
28      document.writeln("<b>accountBalance: </b>" + accountBalance + "<br>");
29      document.writeln("<b>Male: </b>" + male + "<br>");
30
31      // Test join()
32      document.writeln("=========================================<br>");
        varjoined = male.join("<-->");
33      document.writeln("<b>Joined: </b>" +joined + "<br>");
34
35
36      // Test Capitalize
37      document.writeln("=========================================<br>");
        document.writeln("<b>Capitalize: </b>" + capitalize("hello") + "<br>");
38
39      // Test map()
40      document.writeln("=========================================<br>");
41      var capitalized = male.map(capitalize);
42      document.writeln("<b>Map(Capitalize): </b>" + capitalized + "<br>");

43      var balancePlusInterest = accountBalance.map(addInterest);
44      document.writeln("<b> Map(addInterest) </b>" +balancePlusInterest +
        "<br>");
45
46      // Test pop()
47      document.writeln("=========================================<br>");
48      var popedElement = male.pop();
49      document.writeln("<b>Male.pop(): </b>" + popedElement + "<br>");
50
51      // Test push()
52      document.writeln("=========================================<br>");
53      male.push("Cyrus");
54      document.writeln("<b>Male.push(): </b>" + male + "<br>");
55
56  </script>
```



ASSOCIATIVE ARRAYS

- Associative array is an array that uses a “string” (instead of a number) as the index to the elements of an array.

- **Example:**

```
var testScore = []; // Create an empty array

testScore["Bob"] = 89;
testScore["Mary"] = 95;

document.write(testScore["Bob"]);

// The array. length will print a zero because apparently
// the length only counts the array with numeric index!!
document.write(testScore.length + "<br />");


// To examine the elements of the array
for(var student in testScore)
{
        document.write(student + " : ");
        document.write(testScore[student] + "<br />");
}
```

AN INTRODUCTION TO WEB PROGRAMMING

Dr. Hossein Hakimzadeh
Department of Computer and Information Sciences
Indiana University
South Bend, IN

Java Script D

LOOPS

- **While Loop**

```
var number=0;
while (number<5){
  document.write(number+"<br>");
  number++;
}
```

- **Do-While Loop**

```
var number=0;
do{
  document.write(number+"<br>");
  number++;
}
while (number<5) ;
```

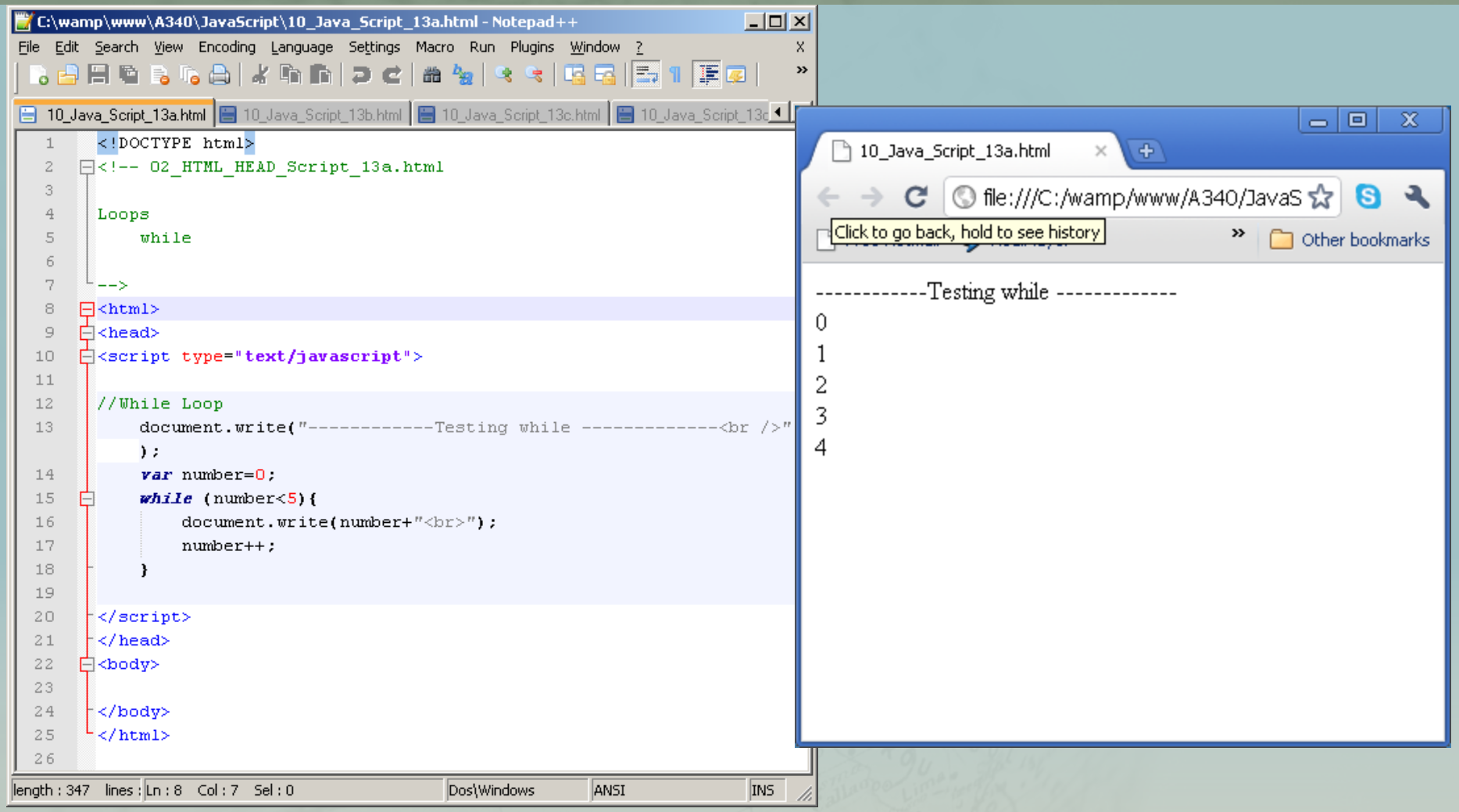
- **For Loop**

```
for (var i=0; i<3; i++){
  document.write("This text is repeated three times<br>");
}
```

- **For –in Loop**

```
var userprofile={name:'George', age:30, sex:'male', getage:function(){return this.age}};

for (var attr in userprofile){
  document.write("<b>"+attr+":</b> "+userprofile[attr]+"<br />");
}
```

WHILE LOOP


C:\wamp\www\A340\JavaScript\10_Java_Script_13a.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
10_Java_Script_13a.html
10_Java_Script_13b.html
10_Java_Script_13c.html
1 <!DOCTYPE html>
2 <!-- 02_HTML_HEAD_Script_13a.html
3
4 Loops
5 while
6
7 -->
8 <html>
9 <head>
10 <script type="text/javascript">
11
12 //While Loop
13 document.write("------------Testing while -------------
");
14 var number=0;
15 while (number<5){
16 document.write(number+"<br>");
17 number++;
18 }
19
20 </script>
21 </head>
22 <body>
23
24 </body>
25 </html>
26
length : 347 lines : Ln : 8 Col : 7 Sel : 0 Dos\Windows ANSI INS
10_Java_Script_13a.html
file:///C:/wamp/www/A340/JavaS
Click to go back, hold to see history
Other bookmarks
------------Testing while -------------
0
1
2
3
4

DO-WHILE LOOP

C:\wamp\www\A340\JavaScript\10_Java_Script_13b.html - Notepad++

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

10_Java_Script_13a.html | 10_Java_Script_13b.html | 10_Java_Script_13c.html | 10_Java_Script_13c

```
<!DOCTYPE html>
<!-- 02_HTML_HEAD_Script_13b.html

Loops
    do-while

-->
<html>
<head>
<script type="text/javascript">

//Do-While Loop
    document.write("------------Testing do-while ----------<br
    />");
    var number=0;
    do{
        document.write(number+"<br>");
        number++;
    } while (number<5) ;


</script>
</head>
<body>

</body>
</html>
```

length : 362 lines : Ln : 1 Col : 1 Sel : 0 | Dos\Windows | ANSI | INS

10_Java... | 10_Java...

file:///C:/wamp/www/A:

Free Hotmail | RealPlayer | Other bookmarks

```
------------Testing do-while ----------
0
1
2
3
4
```

FOR LOOP

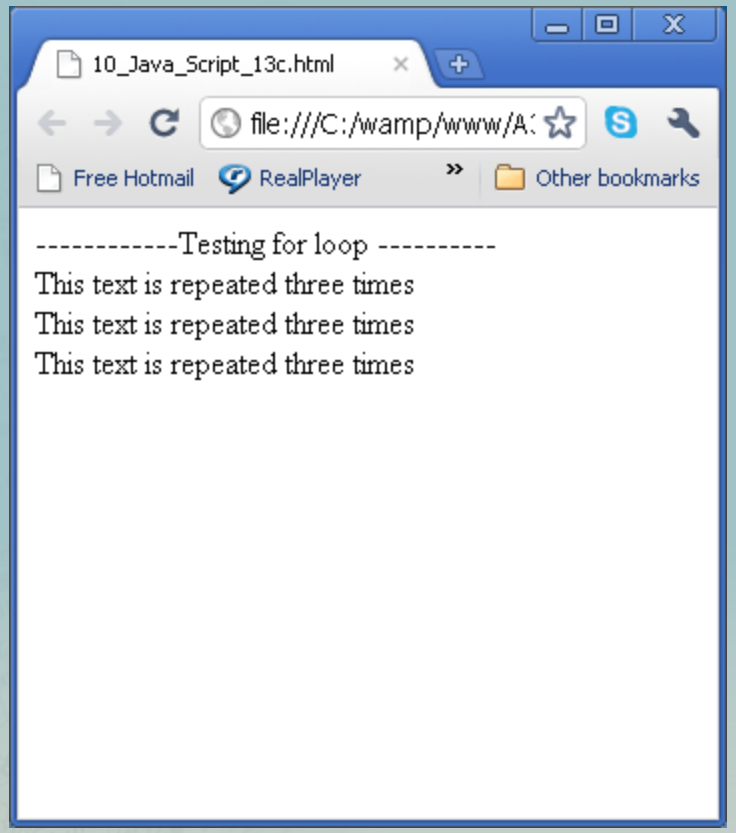
C:\wamp\www\A340\JavaScript\10_Java_Script_13c.html - Notepad++

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

10_Java_Script_13a.html | 10_Java_Script_13b.html | 10_Java_Script_13c.html | 10_Java_Script_13d.h

```
<!DOCTYPE html>
<!-- 02_HTML_HEAD_Script_13c.html

Loops

    for

-->
<html>
<head>
<script type="text/javascript">

//For Loop
    document.write("------------Testing for loop ----------<br />");
    for (var i=0; i<3; i++){
        document.write("This text is repeated three times<br>");
    }

</script>
</head>
<body>

</body>
</html>
```

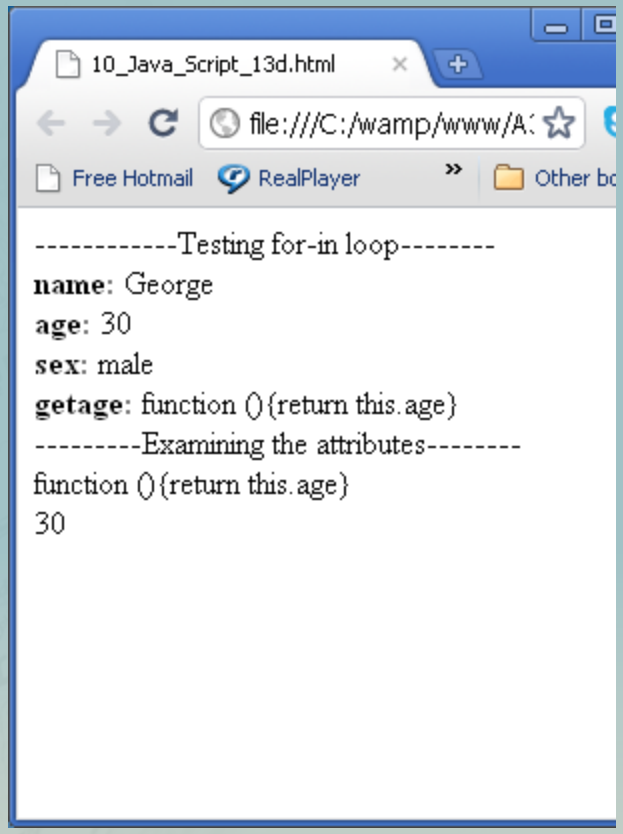
length : 350 lines : 2 Ln : 1 Col : 1 Sel : 0 Dos\Windows ANSI INS



FOR-IN LOOP

10_Java_Script_13b.html | 10_Java_Script_13c.html | 10_Java_Script_13d.html | 10_Java_Script_13e.html

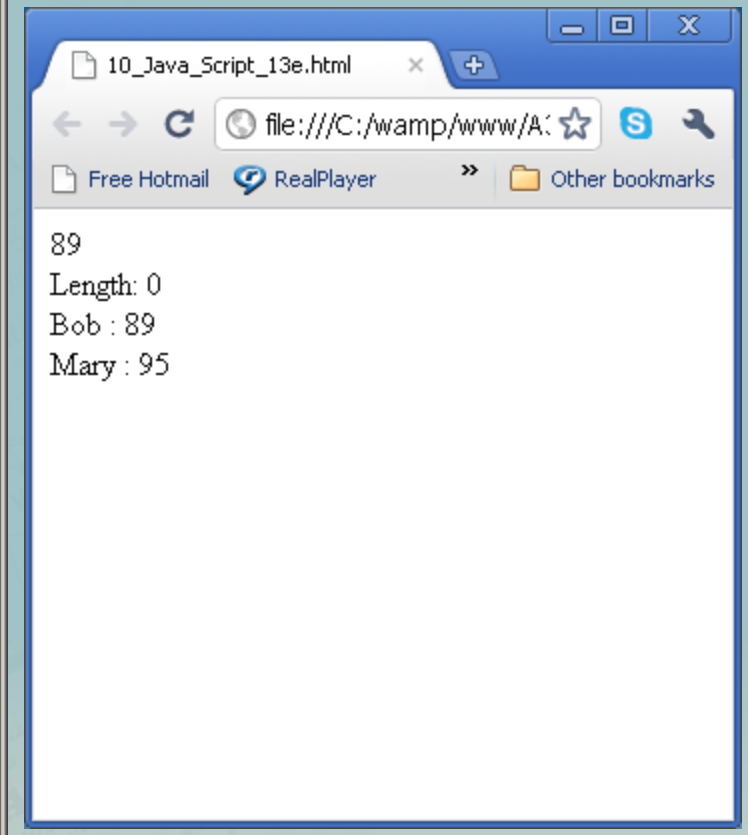
```
Loops
    for-in

-->
<html>
<head>
<script type="text/javascript">


//For-in Loop
    document.write("------------Testing for-in loop--------<br />");
    var userprofile=    {name:'George',
                         age:30,
                         sex:'male',
                         getage:function(){return this.age}
                        };
    for (var attr in userprofile){
        document.write("<b>"+attr+":</b> " +
                       userprofile[attr]  +
                       "<br />");

    }

    document.write("---------Examining the attributes--------<br />");
    document.write(userprofile.getage   + "<br />");    // display the element
    document.write(userprofile.getage() + "<br />");    // execute the element

</script>
</head>
<body>

</body>
```



ASSOCIATIVE ARRAYS AND FOR-IN LOOP

10_Java_Script_13c.html | 10_Java_Script_13d.html | 10_Java_Script_13e.html

```
4   Use of for-in loops and Associative Arrays
5
6   -->
7   <html>
8   <head>
9   <script type="text/javascript">
10
11
12      var testScore = [];          // Create an empty array
13
14      testScore["Bob"] = 89;
15      testScore["Mary"] = 95;
16
17      document.write(testScore["Bob"]);
18
19      // The array.length will print a zero because apparently
20      // the length only counts the array with numeric index!!
21      document.write(testScore.length + "<br />");
22
23
24      // To examine the elements of the array
25      for(var student in testScore)
26      {
27          document.write(student + " : ");
28          document.write(testScore[student] + "<br />");
29      }
30
31
32  </script>
33  </head>
34  <body>
35
36  </body>
```



CONDITIONAL

- Same as C and C++
- If Statement

```
if (expression)
    statement1;
else if (expression2)
    statement2;
else
    statement3;
```

- Switch Statement

```
switch (expression){
    case label1:
            statement1
            break
    case label2:
            statement2
            break
    default: statement3;
}
```

MODULES

- JavaScript has several built-in objects. Each of these objects have several **built-in Functions**:

Built-in Objects:

- Array
- Boolean
- Date
- Math
- Number
- String
- RegExp
- Global

Some Built-in Functions:

- alert();
- confirm();
- prompt();

- parseInt(x);
- parseFloat(x);

- Math.max(x,y);
- Math.min(x,y);
- Math.pow();
- Math.random();

- Date();
- toString()
- charAt()
- toLowerCase()
- toUpperCase()

- escape(), unescape()

http://www.tutorialspoint.com/javascript/javascript_builtin_functions.htm

http://www.javascriptkit.com/jsref/math.shtml

MODULES

- **User-defined Functions:**

```
function getArea(w, h)
{
        var  area = w*h;
        return area;
}
```

```
document.writeln(getArea(3,5)); //call the function
```

MODULES

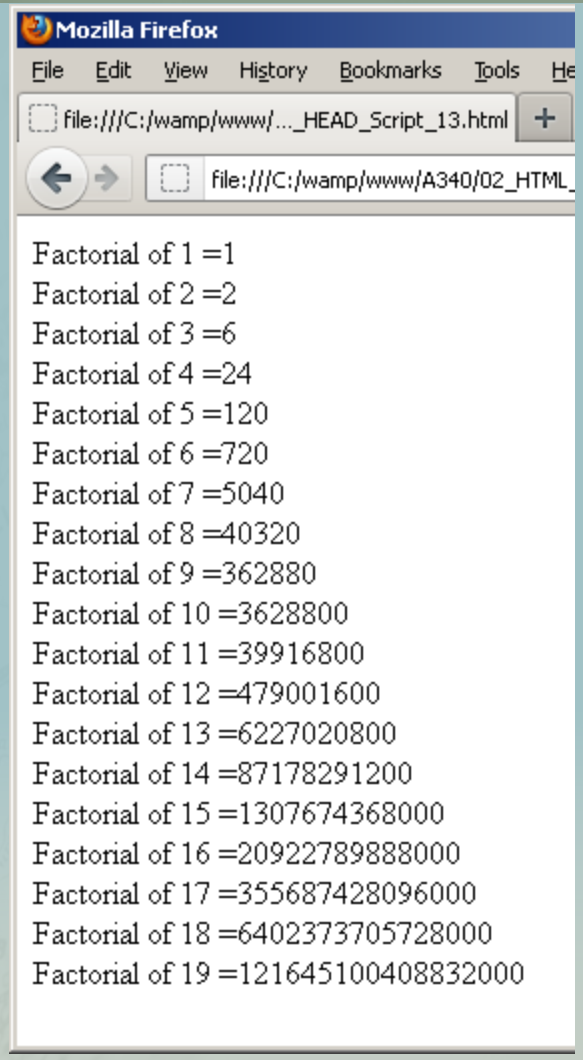
- Recursive Functions:

```
function factorial(number)
{
        if (number <=1)         //base case
                return 1;
        else
                return (number * factorial(number-1));
}

alert(factorial(5)); //call the function
```

RECURSIVE FUNCTIONS

02_HTML_HEAD_Script_13.html

```
<!DOCTYPE html>
<!-- 02_HTML_HEAD_Script_13.html

Recursive functions

-->
<html>
<head>
<script type="text/javascript">


    //Recursive Functions:
    function factorial(number)
    {
        if (number <=1)      //base case
            return 1;
        else
            return (number * factorial(number-1));
    }


    for (var i=1; i<20; i++)
        document.write("Factorial of " + i + " =" + factorial(i) +"<br>");

</script>
</head>
<body>

</body>
</html>
```



AN INTRODUCTION TO WEB PROGRAMMING

Dr. Hossein Hakimzadeh
Department of Computer and Information Sciences
Indiana University
South Bend, IN

Java Script E

JAVASCRIPT AND ITS RELATION WITH DOM VS. BOM

- **Document Object Model (DOM)**

- The Document Object Model (DOM) is an application programming interface (API) for HTML as well as XML.

- The DOM organizes the entire web page as a document composed of a hierarchy of nodes like a tree structure and using the DOM API, nodes can be removed, added, and replaced.

- **DOM allows the developer to manipulate the document.**

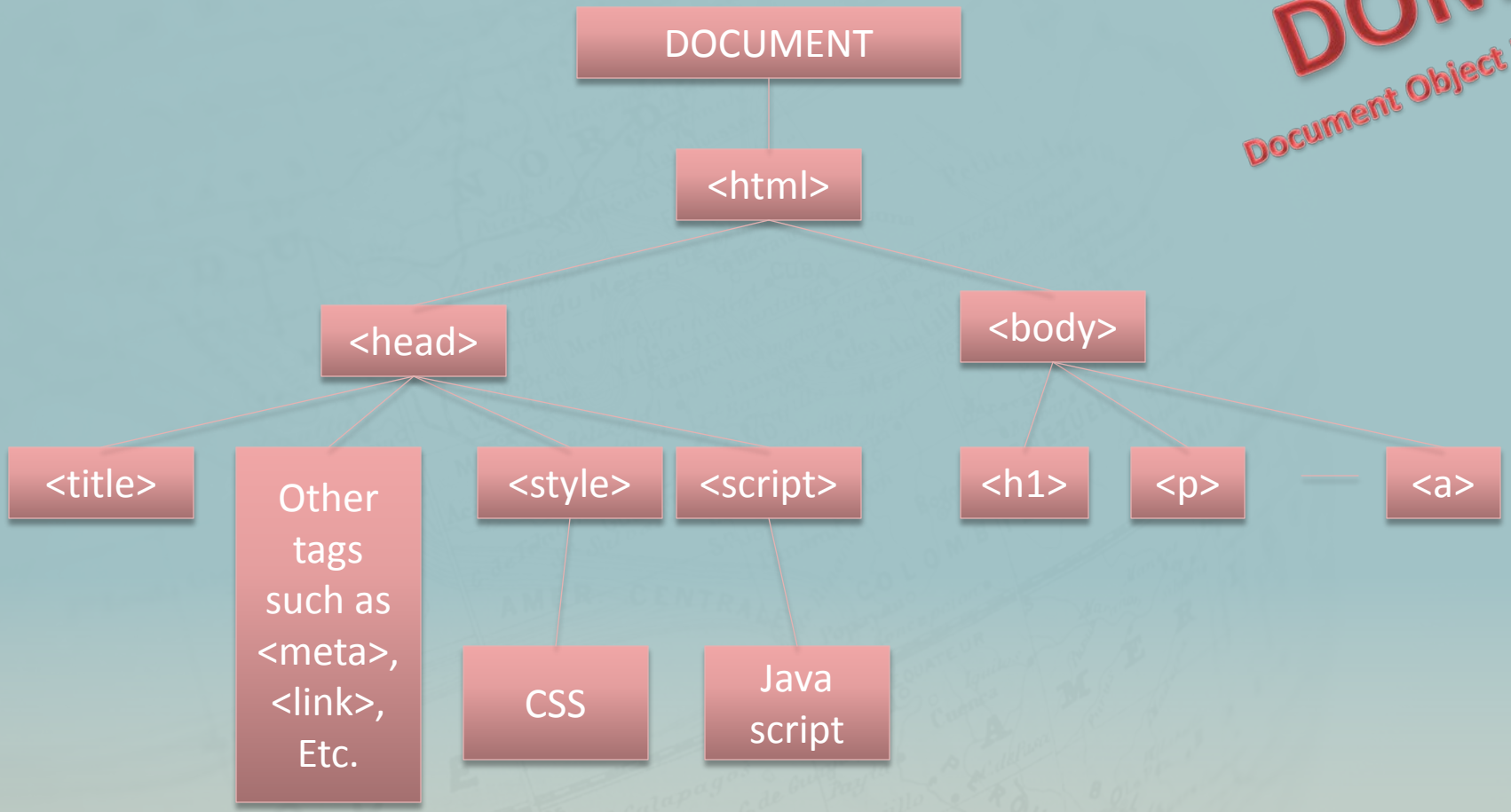
Browser Object Model (BOM)

- Browsers feature a Browser Object Model (BOM) that allows access and manipulation of the browser window. For example (browser history, location, navigator, and screen)

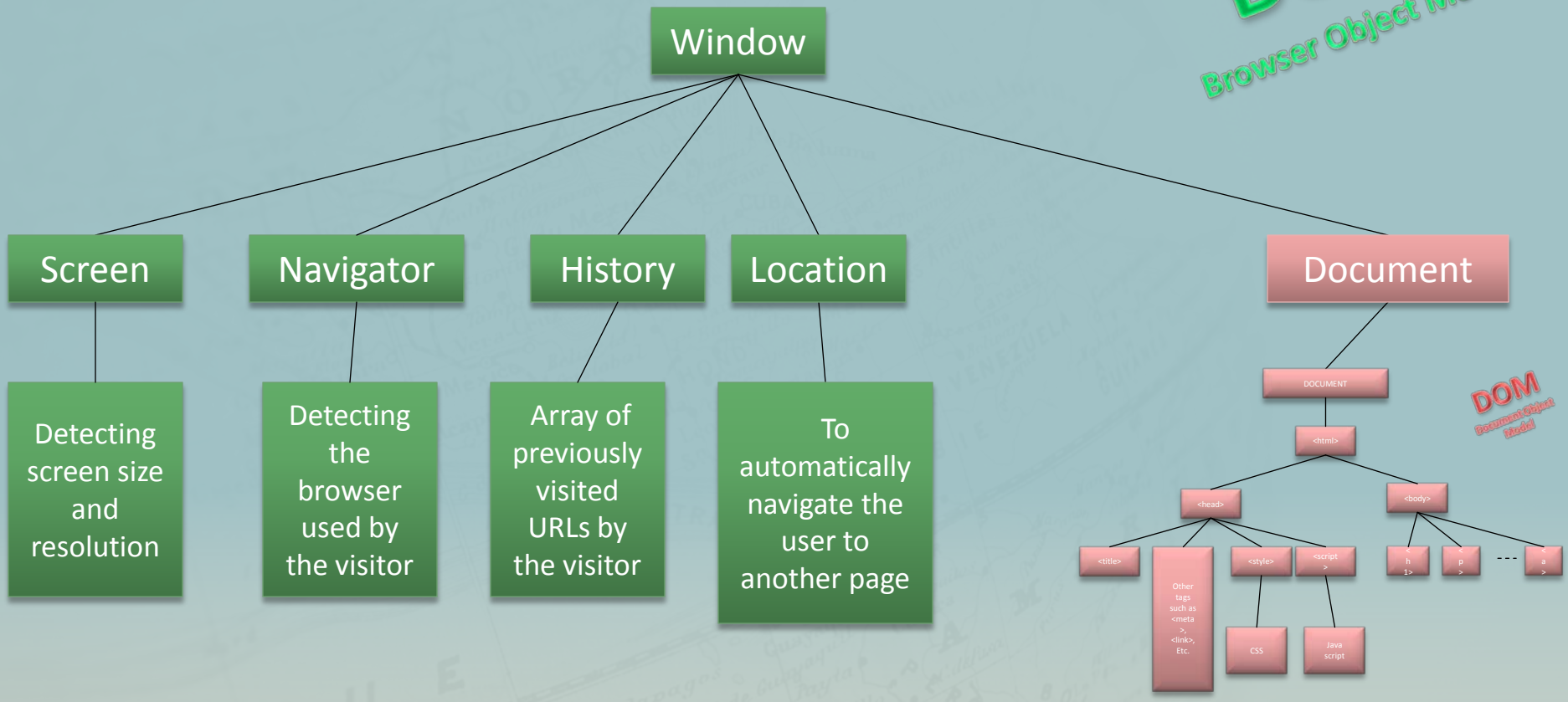
- Because no standards exist for the BOM, each browser has its own implementation.

- **BOM allows the developer to manipulate the browser window.**

THE STRUCTURE OF AN HTML DOCUMENT



THE STRUCTURE OF BROWSER OBJECT



http://www.javascriptkit.com/jsref/window.shtml

INTERACTING WITH BOM

- **Screen:** contains information about the visitor's screen.
  - (height, width, colorDepth, etc)
- **Window:** represents an open window in a browser.
  - (open, close, name, length, width, parent, screenleft, screentop, alert(), prompt(), print(), moveTo(),moveBy(), scrollBy(), scrollTo(), blur(), setTimeout(), etc)
- **Navigator:** contains information about the browser.
  - (appName, CookiesEnabled, JavaEnabled, etc)
- **History:** contains the URLs visited by the user (within a browser window).
  - (length, back(), forward(), go(), etc.)
- **Location:** contains information about the current URL.
  - (host, hostname, href, port, protocal, assign(), reload(), replace())

http://www.w3schools.com/jsref/obj_window.asp

INTERACTING WITH DOM

- The Document object is the root of a document tree. It gives us access to the document's data elements.
- Since element nodes, text nodes, attributes, comments, etc. cannot exist outside the document, the Document object contains methods to create and access these objects.
- **Properties:** doctype, documentURI, inputEncoding, xmlEncoding, anchors, forms, images, links, referrer, title, URL, domain, lastmodified, etc.
- **Methods:** open(), close(), write(), writeln(), getElementByName(), getElementById(), getElementByTagName(), renameNode(), setAttribute(), getAttribute(), removeAttribute(), etc.
- **Events:** load, unload, resize, scroll, click, dblclick, mousedown, mouseup, mouseover, mouseover, mouseout, keydown, keyup, keypress, etc.

http://en.wikipedia.org/wiki/DOM_events

http://www.w3schools.com/jsref/dom_obj_node.asp

SCREEN / MONITOR OBJECT

BOM
Browser Object Model

```
document.write("<b>Total Width of the screen (including the Windows Taskbar):</b> " + screen.width + "<br />");
document.write("<b>Total Height of the screen (including the Windows Taskbar):</b> " + screen.height + "<br />");

document.write("<b>Width of the screen (excluding the Windows Taskbar):</b> " + screen.availWidth + "<br />");
document.write("<b>Height of the screen (excluding the Windows Taskbar):</b> " + screen.availHeight + "<br />");

document.write("<b>Bit depth of the color palette for displaying images:</b> " + screen.colorDepth + "<br />");
```

WINDOW /BROWSER OBJECT

BOM
Browser Object Model

```
document.write("<b>X coordinate of the window relative to the screen:</b> " +  window.screenX + "<br />");
document.write("<b>Y coordinate of the window relative to the screen:</b> " +  window.screenY + "<br />");
document.write("<b>Inner Width of a window's content area:</b> " +  window.innerWidth + "<br />");
document.write("<b>Inner Height of a window's content area:</b> " +  window.innerHeight + "<br />");
document.write("<b>Outer width of a window, including toolbars/scrollbars:</b> " +  window.outerWidth + "<br />");
document.write("<b>Outer height of a window, including toolbars/scrollbars:</b> " +  window.outerHeight + "<br />");
document.write("<b>Current URL:</b> " + window.location + "<br />");
```

WINDOW OBJECT (CONTINUED…)

- **To create a new Browser Window:**
  - var myWin = window.open('', '_blank', 'width=100, height=400'); //make sure the pop-up blocker is off
- **To write to that window:**
  - myWin.document.write("Hello");
- **To position and size the window the window:**
  - myWin.moveTo(0,0); // not consistent (works in IE)
  - myWin.resizeTo(500,500); // not consistent (works in IE)
- **To focus on the window:**
  - myWin.focus(); // not consistent (works in IE)
- **Creating a Timer Object: (to do something at a later time)**
  - myWin.document.write("....close in 15 seconds...");
  - myWin.setTimeout(function() {myWin.close()}, 15000);// close window after 15 seconds
- **Creating a Timer Object: (to repeat something on regular intervals)**
  - myWin.setInterval(function() {displayClock(myWin)}, 1000); // run the display clock function every second.

WINDOW OBJECT (CONTINUED…)

- **To Scroll the Browser Window:**
  - window.scrollTo(x,y);

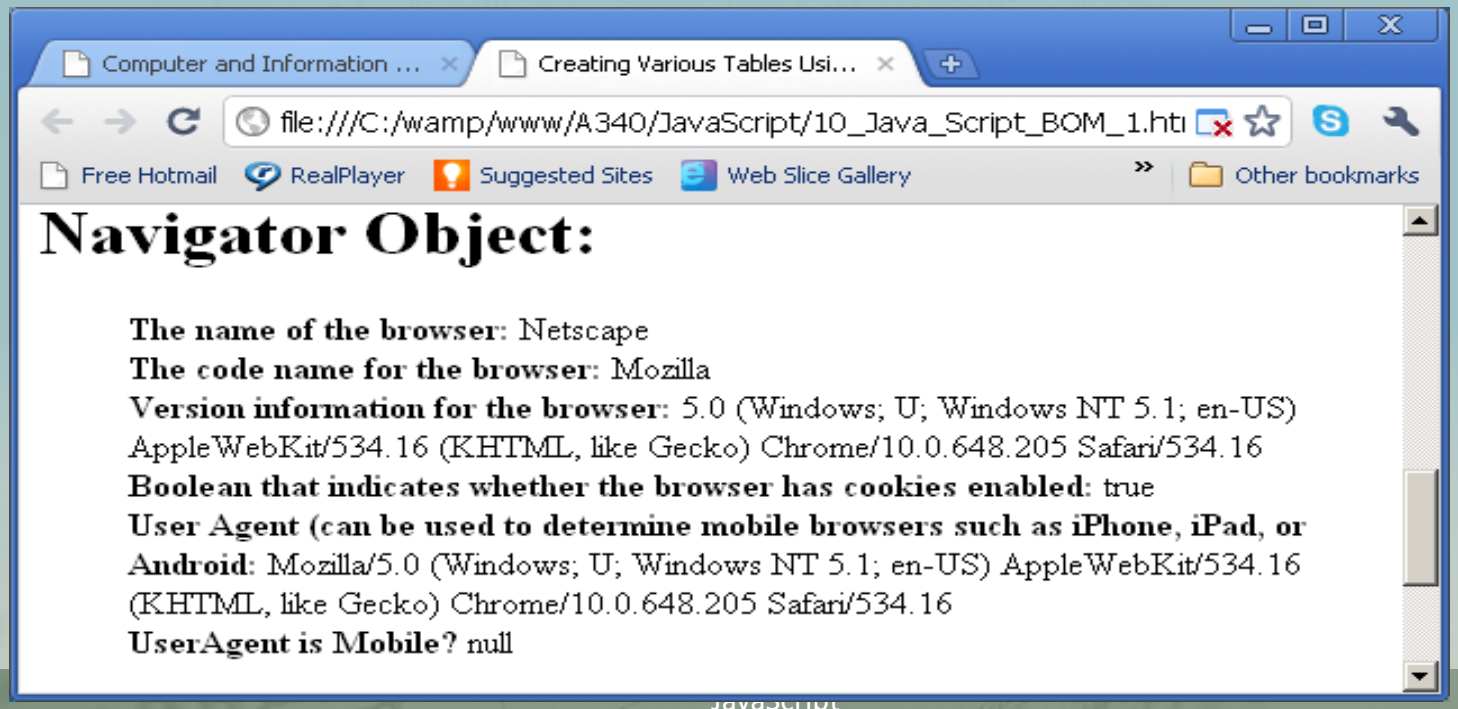
AN INTRODUCTION TO WEB PROGRAMMING

Dr. Hossein Hakimzadeh
Department of Computer and Information Sciences
Indiana University
South Bend, IN

Java Script E1

NAVIGATOR OBJECT


```
document.write("<b>The name of the browser:</b> " + navigator.appName  + "<br />");
document.write("<b>The code name for the browser:</b> " + navigator.appCodeName  + "<br />");
document.write("<b>Version information for the browser:</b> " + navigator.appVersion  + "<br />");
document.write("<b>Boolean that indicates whether the browser has cookies enabled:</b> " + navigator.cookieEnabled + "<br />");
document.write("<b>User Agent (can be used to determine mobile browsers such as iPhone, iPad, or Android:</b> " +
navigator.userAgent + "<br />");                //returns true if user is using one of the following mobile browsers

var isMobile=navigator.userAgent.match(/(iPad)|(iPhone)|(iPod)|(android)|(webOS)/i);
document.write("<b>UserAgent is Mobile?</b> " + isMobile  + "<br />");
```

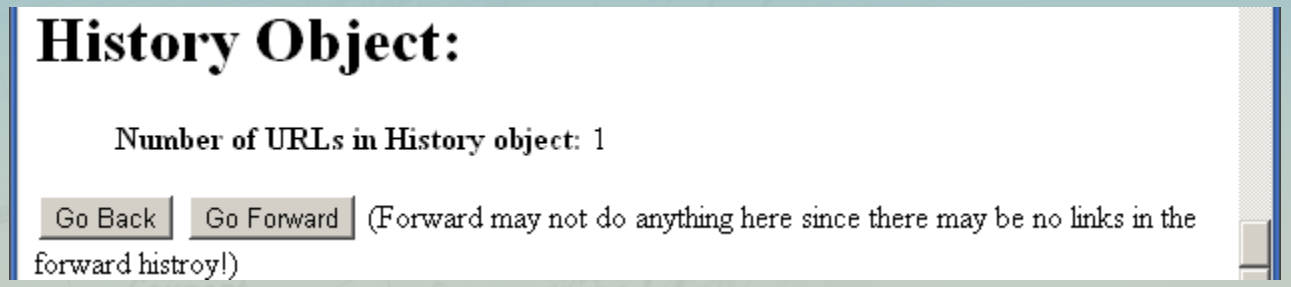


HISTORY OBJECT

- **To get the length:**
  - document.write("Number of URLs in History object: " + history.length + "<br />");
- **To move Backward or Forward:**
  - history.back();
  - history.forward() ;

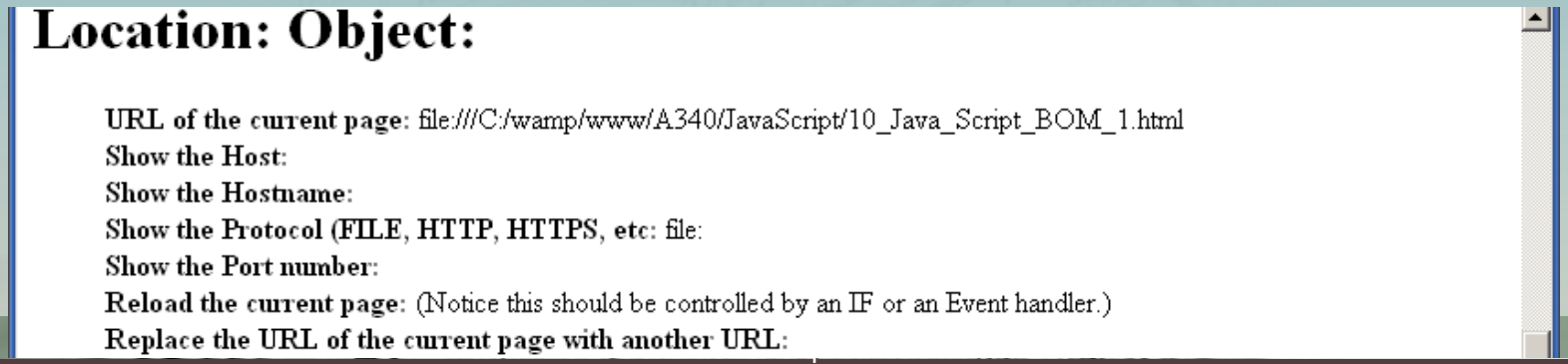
```
function goBack()
{
        window.history.go(-1);
}
function goForward()
{
        window.history.go(+1);
}
```

```
<body>
<input type="button" value="Go Back" onclick="goBack()">
<input type="button" value="Go Forward" onclick="goForward()"> (Forward may not do anything here since there may be no links in the forward history!)
</body>
```



LOCATION OBJECT

```
document.write("<b>URL of the current page:</b> " + location.href + "<br />");
document.write("<b>Show the Host:</b> "  + location.host + "<br />");
document.write("<b>Show the Hostname:</b> "  + location.hostname + "<br />");
document.write("<b>Show the Protocol (FILE, HTTP, HTTPS, etc:)</b> "  +
                                                location.protocol + "<br />");

document.write("<b>Show the Port number:</b> "  + location.port + "<br />");

document.write("<b>Reload the current page:</b> (Notice this should be controlled by an IF
    statement or an Event handler.) "  + "<br />");
    //location.reload();

document.write("<b>Replace the URL of the current page with another URL:</b> "  + "<br />");
    //location.replace('http://www.iusb.edu');
```

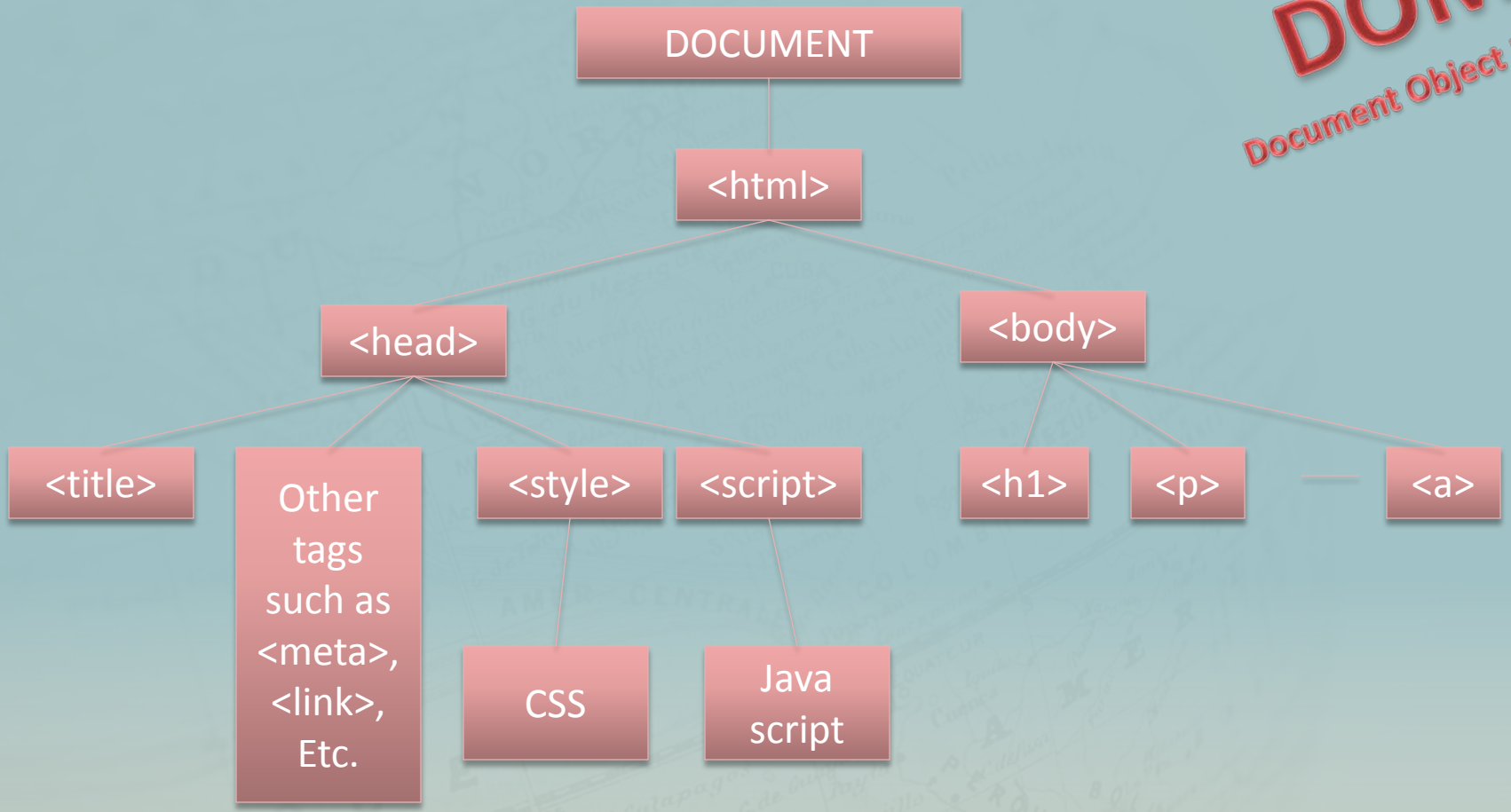


AN INTRODUCTION TO WEB PROGRAMMING

Dr. Hossein Hakimzadeh
Department of Computer and Information Sciences
Indiana University
South Bend, IN

Java Script F

THE STRUCTURE OF AN HTML DOCUMENT



DOM

Document Object Model

INTERACTING WITH DOM

- The Document object is the root of a document tree. It gives us access to the document's data elements.
- Since element nodes, text nodes, attributes, comments, etc. cannot exist outside the document, the Document object contains methods to create and access these objects.
- **Properties:** doctype, documentURI, inputEncoding, xmlEncoding, anchors, forms, images, links, referrer, title, URL, domain, lastmodified, etc.
- **Methods:** open(), close(), write(), writeln(), getElementByName(), getElementById(), getElementByTagName(), renameNode(), setAttribute(), getAttribute(), removeAttribute(), etc.
- **Events:** load, unload, resize, scroll, click, dblclick, mousedown, mouseup, mouseover, mouseover, mouseout, keydown, keyup, keypress, etc.

http://en.wikipedia.org/wiki/DOM_events

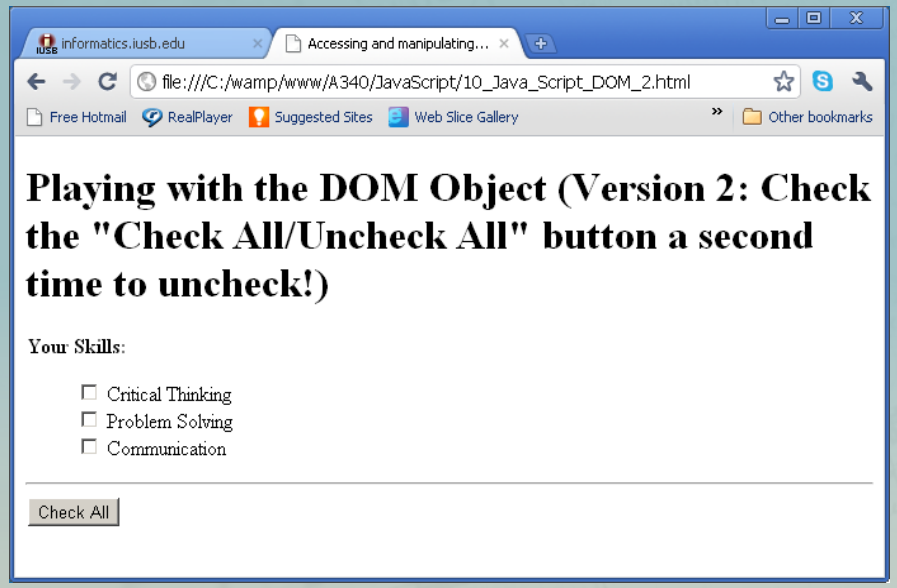
http://www.w3schools.com/jsref/dom_obj_node.asp

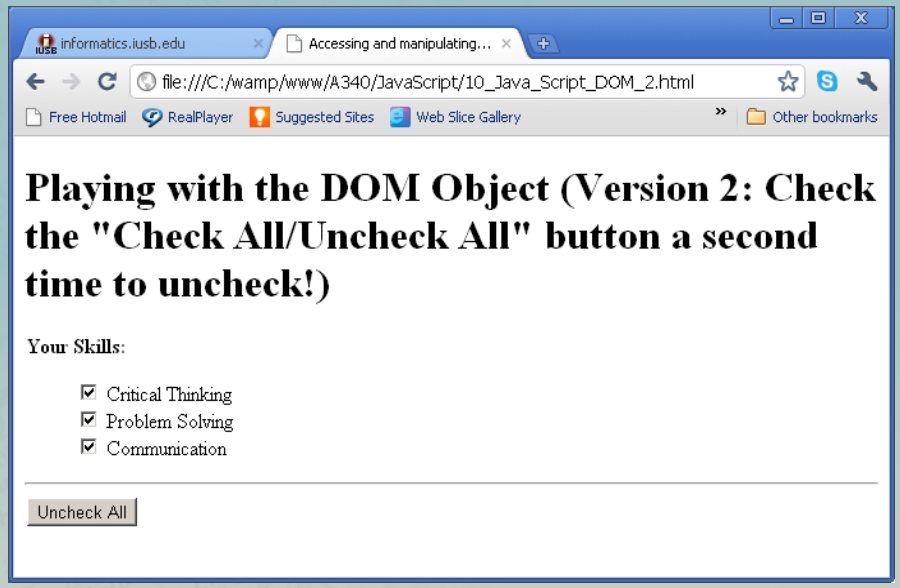
DOM OBJECT

- document.getElementById ()

DOM OBJECT

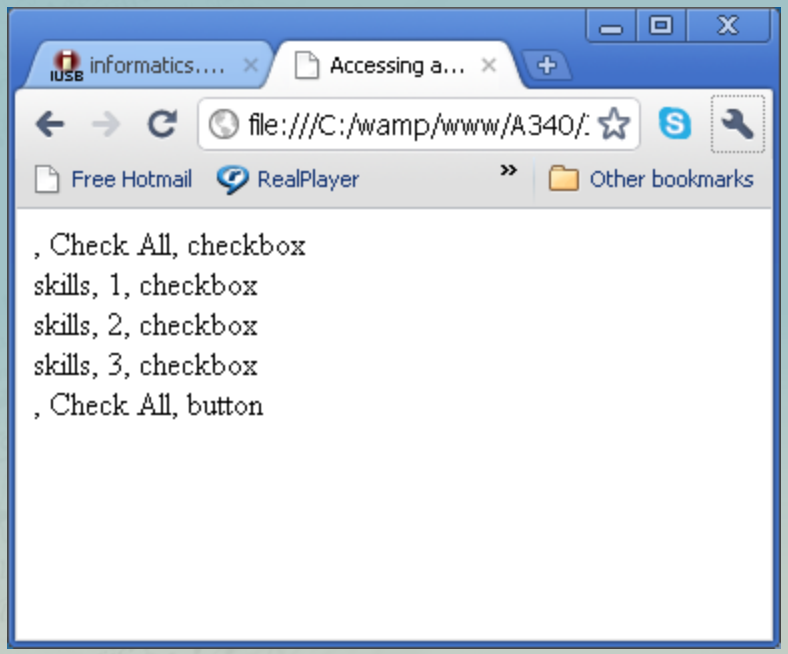
```
var checkAllBtn = document.getElementById("ckallbtn");
checkAllBtn.value = "Uncheck All";
```





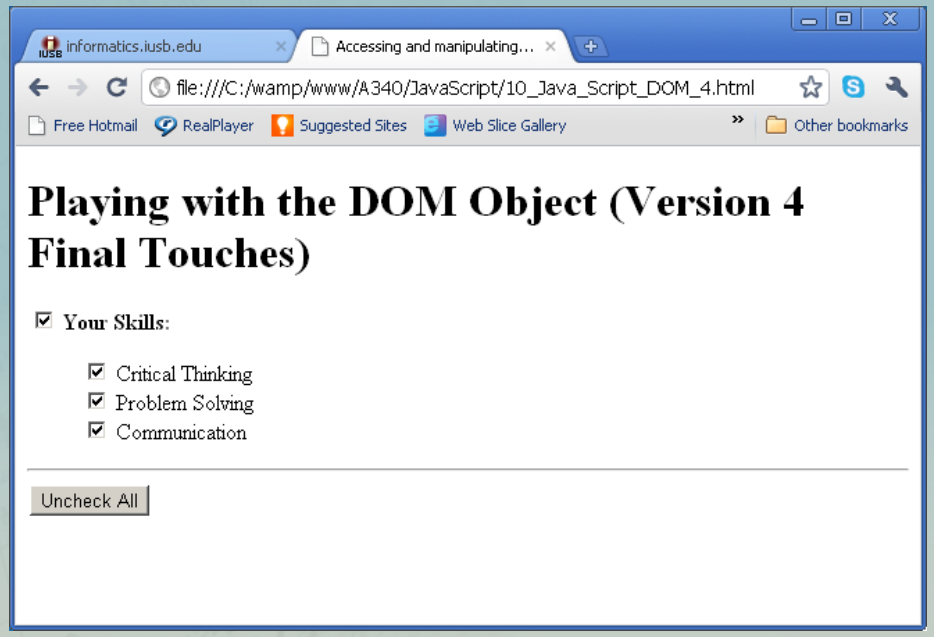
DOM OBJECT

```
// Get the form object
var theForm = document.getElementById("form1");

// Get the elements within the form
var theElements = theForm.elements;
```



DOM OBJECT

```
// Get the form object
var theForm = document.getElementById("form1");

// Get the elements within the form
var theElements = theForm.elements;
```



DOM OBJECT

- Two Dimensional Arrays
- Changing the internal attributes of an HTML table.
- Changing the background color, changing the format format.