JavaScript
Katy Luchini
Michigan State University
http://www.cps.msu.edu/~luchinik/

Part I
JavaScript's History

• Wanted to improve interaction on the web
• Wanted the option of client-side applications
• Scripting was the obvious solution
• Netscape started developing LiveScript to answer these questions
• Once Java was introduced, Netscape began working with Sun on JavaScript

History, Cont’d

• Java accomplished some of the client-side tasks, but separate from HTML
• JavaScript is less cumbersome, fully integrated into HTML
• Microsoft got involved with its own version, JScript, supported by IE
• Finally turned the language over to ECMA to develop international standards

Key JavaScript Facts

• Embedded in HTML
  – JavaScripts are downloaded by the browser as part of the webpage
  – Can also be separate, executable files, but that’s rare
• Depends on the browser to run the applications ("client-side scripting")

JavaScript vs. Java

• JavaScript is NOT Java!
  – Java is a separate application
  – The HTML tag <APPLET> tells the browser where to look for the separate Java file
• JavaScript is an integrated application
  – JavaScript uses the <SCRIPT> tag in HTML to tell the browser how to interpret the script

Popular Uses for JavaScript

• Dynamic HTML
  – Special effects, animation, banners
• Client side data validation
  – Make sure user has entered some information before passing it to the server to be validated
  (passwords, forms, etc.)
More Uses

• Provide interactivity without CGI (Common Gateway Interface) scripts or Java – Makes forms, searches, and other types of web interaction faster, easier, and more reliable (doesn't crash machines as often as Java does!)
• Communication within browser windows – Let frames "talk" to each other so you can change information in one frame without having to reload the entire webpage
  – "MouseOver ()" actions respond to user actions

JavaScript and HTML

• Two ways to incorporate JavaScript
  – Include the script within <HTML> document
  – Have a separate file (.js extension) that is called from your <HTML> document (only available in Netscape 3.0+)

<SCRIPT> Tags

• <SCRIPT> tags
  – These are the HTML identifiers that tell the browser that the page is using a scripting language.
  – The "Language=JavaScript" identifier tells the browser that JavaScript is the specific language being used.

Functions

• JavaScript separated into two parts of the <HTML> document
  – Functions are defined in the <HEAD>
  – Functions are called in the <BODY>
• Comments (<!--  -->) used to hide JavaScript code from older browsers that can't support it
  – In older browsers, the JavaScript code is completely ignored

Example 1:

JavaScript Functions

• Calling a JavaScript function from within an HTML document
• Difference between HTML instructions and JavaScript code

<HTML>
<HEAD>
<TITLE>Calling Function Demo</TITLE>
<SCRIPT LANGUAGE= "JavaScript">
funtion  displayMessage () {
document.write("JavaScript functions are easy to use!")
document.write("<BR>")
}
</SCRIPT>
</HEAD>
Go to Exercise 1
<HTML>
<TITLE>User Interaction Demo</TITLE>
<SCRIPT LANGUAGE= "JavaScript">
<!-- begin hiding from older browsers

function Act()
{var Text = document.exampleForm.userText.value;
var ind = document.exampleForm.pickList.selectedIndex;
if( ind == 0 ) return;
else if ( ind == 1 ) showMessageBox( Text );
else if ( ind == 2 ) showWindow( Text );
else if ( ind == 3 ) showStatusBar( Text );
document.exampleForm.pickList.selectedIndex = 0;}

function showMessageBox( Text )
{var CRLF = "\r\n";
alert( "You entered: " + CRLF + CRLF + Text );}

function showWindow( Text )
{newWindow = window.open("","" , "toolbar, menubar ,scrollbars,
resizable,height=300,width=400");
newWindow.document.open();
newWindow.document.write( Text );
newWindow.document.close();}

function showStatusBar( Text )
{window.status = Text;}

function goToURL()
{top.location = "http://www.cps.msu.edu/~luchinin/"}
//-->
</SCRIPT>
</HEAD>
<BODY>

<FORM NAME="exampleForm">
<I>Enter Text Here: </I><BR>
<INPUT TYPE="text" NAME="userText">
<BR>
<BR>
<I>Select Display Option:</I><BR>
<SELECT NAME="pickList" SIZE=1 onChange ="Act()">
<OPTION VALUE="">(Select One)
<OPTION VALUE="">Message Box
<OPTION VALUE="">Window
<OPTION VALUE="">Status Bar
</SELECT>
<BR>
<BR>
<INPUT TYPE="Button" Value="Go To Lecture" onClick ="goToURL()">
</FORM>
</BODY>
</HTML>
Exercise

• Look at this example online
• Go to the URL by clicking the button in the next slide
• Try the different display types (remember to close the new boxes and windows, and refresh the page if necessary)
• Look at the source code (find a quiz answer there)
• Click on the "Go To Lecture" button on the webpage and come back for part 3 of the lecture when you're finished

Go to Exercise II

JavaScript

Katy Luchini
Michigan State University
http://www.cps.msu.edu/~luchinik/

Part III

Key Points in Example 2

• Act() function calls one of the other three functions depending on which index value is selected
• showMessageBox() displays the text in a message box
• showWindow() displays the text in another browser window
  – Note that the string of numbers and letters after the window.open command describes which elements (menu bars, etc.) should appear in the new window

More Key Features

• showStatusBar() displays the text in the status bar at the bottom of the browser window
• goToURL() goes to a specified URL address when the "Go To Lecture" button is pushed

Lecture Summary

• JavaScript is an integrated approach to HTML
• Provides for client-side applications
• Faster and more reliable than CGI or Java
• Two ways to incorporate JavaScript into HTML
  – <SCRIPT> tag
  – Separate file with .js extension
Summary, Cont’d

- JavaScript functions are distinct from HTML
  - Functions defined in <HEAD>
  - Functions called in <BODY>

Variety of uses for JavaScript
- User Interaction (web forms, games, mouseOver())
- Communication (inter-frame, status bar)
- Data validation (forms, passwords)

Web Resources
- Self-contained JavaScript code easily accessible on the web
- Lots of “cut and paste” sites
- Can also “View | Page Source” on pages supporting JavaScript
- Click on the button on the next slide to go to a page of JavaScript resources

Go to JavaScript