

CS120

Introduction to Web Site Development

Lecture 8 - JavaScript II.

Gabor Kondas

California State University, Los Angeles

(Based on slides by John Hurley)

JavaScript Comments

- Comments are ignored by the web browser
- There are two types of comments
- Single Line

`// comment`

- Multiple Line

`/* comment`

`comment`

`comment */`

Comments Example

Example:

```
<script type="text/javascript">  
    // this is a comment  
  
    /*  
    this is a comment as well  
    it spans multiple lines  
    */  
</script>
```

JavaScript Functions

- A function is a key programming construct.
 - A statement is the smallest unit of execution in a programming language
 - A function executes zero or more statements
 - Functions can be reused

JavaScript Functions

- A function in JavaScript is like a function in math
- This is how we do it in math
 - The function definition (math):
 - $f(x) = x^2$
 - Evaluating (calling) a function (math)
 - $f(2) = 4$
 - $f(3) = 9$
 - $f(4) = 16$

JavaScript Functions

- The function definition (JavaScript):
 - `function square(input) // x is the argument`
`{`
 `return input*input; // multiply x times x`
`}`
- Evaluating (calling) a function (JavaScript)
 - `var x = square(2); // evaluates to 4`
 - `var y = square(3); // evaluates to 9`
 - `var z = square(4); // evaluates to 16`
- If you have a return statement in your function, you may call the function and assign the result to a variable as in the example above

JavaScript Functions

Every JavaScript function declaration has the following syntax:

- The function definition (JavaScript):

```
function fname(arg1, arg2)
{
    statement-list
}
```

JavaScript Functions

JavaScript functions can have zero arguments

```
function hey()  
{  
    window.alert("Hey!");  
    return "Said \"Hey!\"";  
}
```

JavaScript functions do not have to return anything (they can just do one or more things)

```
function hey()  
{  
    window.alert("Hey!");  
    window.alert("Hey again!");  
}
```


JavaScript Functions

Anytime you have a JavaScript function that doesn't return something (doesn't have the return statement), don't assign it to variable

```
function hey1()  
{  
    return 0;  
}
```

```
function hey2()  
{  
}
```

```
var x = hey1(); // OK, function hey1 returns 0  
var y = hey2(); // NOT OK! NO RETURN VALUE!
```

JavaScript Functions

You must call a function for it to run;
just defining the function does not make it run

```
<!DOCTYPE html>
<html>
  <head>
    <script type="text/javascript">
      function hey() {
        window.alert("Hey!");
        return ("Said \"Hey!\"");
      }
    </script>

  </head>
  <body>
    <script type = "text/javascript">
      document.write(hey());
    </script>
  </body>
</html>
```

Properties

A property is a characteristic or attribute of an object. Properties generally correspond to html element attributes. Many properties have slightly different names than the corresponding css properties.

- The background color of a web page document
document.bgcolor
- The date the web page file was last modified
document.lastmodified
- The src file of an image object
image1.src

```
<!DOCTYPE html>
<html>
  <head>

  </head>
  <body>
    <script type="text/javascript">
      var colorIn = window.prompt("what is your
      favorite color?");
      changeBG(colorIn);

      function changeBG(color) {
        document.bgColor = color;
        //Change background color

      }
    </script>
  </body>
</html>
```

Objects

An object is a thing or entity.

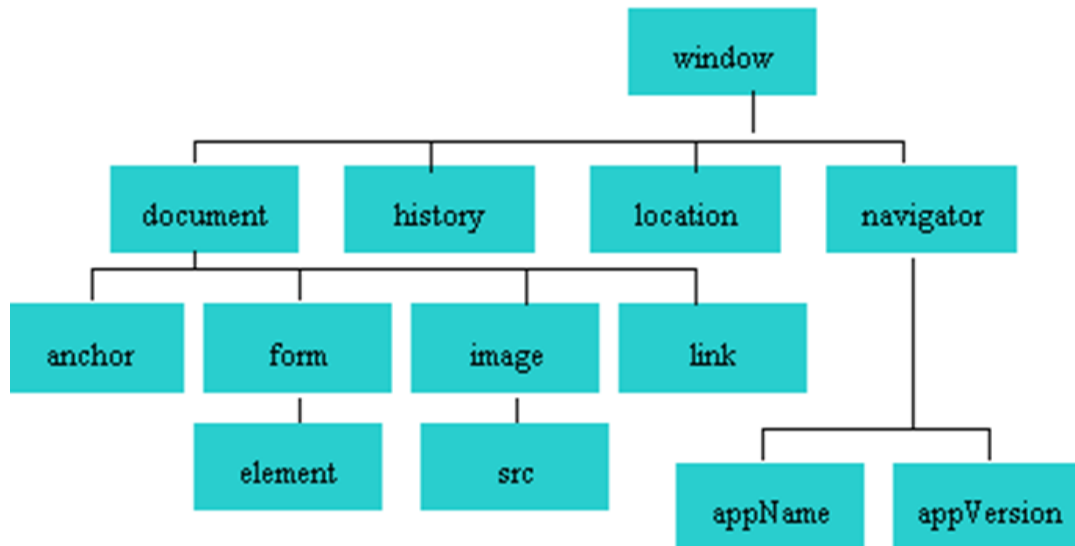
- Browser window
- Submit button
- Web page document

[JS Objects Resources](#)

Document Object Model

- In programming terms, an Application Programming Interface
- A way to manipulate an html document
- DOM allows us to use JavaScript to control various aspects of our documents

Document Object Model (DOM)



- A portion of the DOM is shown at the left.
- Defines every object and element on a web page
- Hierarchical structure
- Accesses page elements and apply styles to page elements

The Document Object

- Every HTML page has a document object associated with it
- The document object gives us access to the other HTML objects in our document, plus a few other things
- The syntax is:
 - `document.attribute`
 - `document.function-name(parameters)`

Read/Write Document Attributes

- `document.title`
 - The title of the HTML document, usually the string that is displayed in the browser's window caption or tab bar
- `document.dir`
 - The general direction of text in the document; can be the string "ltr" or "rtl"
- `document.body`
 - Represents the body of the document; this one will be critical if you continue your study of JS

Read Only Document Attributes

- `document.URL`
 - A string that contains the full pathname of the currently open HTML document
- `document.lastModified`
 - A string that contains the date for which the currently open HTML document was last modified

Document

```
<head>
  <title>My Document</title>
</head>
<body>
  <script type = "text/JavaScript">
    window.alert(document.title);
    document.title = "Godzilla's Lair";
    window.alert(document.title);
  </script>
</body>
```

Document Functions

- `document.write(str)`
 - A function that allows you append HTML code to your document while it is being loaded
- `document.getElementById(id)`
 - Retrieves the HTML object associated with the given HTML id attribute;
- `document.getElementsByTagName(tag)`
 - Retrieves all HTML objects associated with the given element name;

GetElementById

`document.getElementById(id)` returns whatever HTML element object the ID refers to, for example:

- Paragraph
- Table

HTML Element

- Once we get an element using the function `document.getElementById`, we can
 - Change the content (innerHTML) of the element
 - Change the attributes of the element
 - Change the direction of the element
 - Change the CSS class name of the element
 - And much more

Getting HTML Elements By ID

First, note what you get when you display the element retrieved

[Example](#)

HTML Element Attributes

All HTML objects have the following attributes, for which you can change using the HTML DOM

- id
- className
- title
- dir
- innerHTML
 - Inner HTML is the content between the start and end tags of an HTML element

Change CSS Styles Using `getElementById`

- Create an HTML element, say a P element
- Assign an ID to the element
- Assign a CSS class name to the element
- Get a handle to the HTML element using the `document.getElementById(ID)` function
- Assign a new class name to the elements `className` attribute

How to Change Inner HTML

Create an HTML element, say a p element with some content in it

Assign an ID to the element

Get a handle to the HTML element using the `document.getElementById(ID)` function

Assign new content using `handle.innerHTML`

The `innerHTML` in this case is a string that can even contain HTML code!

getElementsByTagName

- Gets an array of all elements of a stated tag type, which you can access using Java-like array notation

getElementsByTagName

```
<body>
<p>Learned discussion of Marcel Proust here</p>
<p>Biography of Cicero here</p>
<script>
    var paragraphs =
        document.getElementsByTagName("p");
    paragraphs[0].innerHTML = "Godzilla was taller";
    paragraphs[1].innerHTML = "Bruce Lee was a better
actor";
</script>
</body>
```

JavaScript and Events

JavaScript can be configured to perform actions when events occur.

- The event name is coded as an attribute of an XHTML tag
- The value of the event attribute contains the JavaScript code

Example:

Display an alert box when the mouse is placed over a hyperlink.

```
<a href="http://www.google.com" onmouseover="alert('Google');">Google </a>
```

Events

Event	Event Handler
click	<code>onclick</code>
load	<code>onload</code>
mouseover	<code>onmouseover</code>
mouseout	<code>onmouseout</code>
submit	<code>onsubmit</code>
unload	<code>onunload</code>

[More](#)

HTML Button

Can be used with event attributes like onclick to trigger Javascript functions.

```
<head>
    <meta charset="utf-8" />
    <title>Demo</title>
    <script lang = "javascript">
function thankuser() {
window.alert("thanks");
}
    </script>
</head>
<body>
    <button onclick = "thankuser();">
        Click Me!
    </button>
</body>
```


Navigator Object

Contains information about the browser

```
<body>  
    <script type = "text/JavaScript">  
        window.alert(navigator.appName);  
    </script>  
</body>
```

JavaScript That Redirects Browser Based on Input

```
<script type="text/javascript">
    var legal = window.confirm("Are you 21 or older?");
    if(legal ==false) {
        window.location = "http://www.google.com";

    }
    else window.location = "http://www.guinness.com";
</script>
</body>
```

JavaScript Debugging(1)

- Check the syntax of the statements
 - Pay close attention to **upper and lower case letters**, spaces, and quotations
- Verify that you have saved the page with your most recent changes
- Verify that you are testing the most recent version of the page (refresh or reload the page)
- Most browsers can display error messages:
 - In Firefox: Select Tools / Error Console
 - Chrome: Tools/Developer Tools

JavaScript Resources

JavaScript Tutorials

<http://www.w3schools.com/JS>

JavaScript Tutorial for the Total Non-Programmer <http://www.webteacher.com/javascript/>

More Beginning JavaScript Tutorials <http://echoecho.com/javascript.htm>

Core JavaScript 1.5 Reference Manual http://www.webreference.com/javascript/reference/core_ref

Creating Accessible JavaScript

<http://www.webaim.org/techniques/javascript>