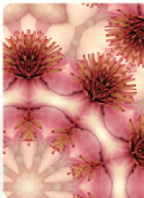


# Class Notes: Tutorial 10 – Programming with JavaScript

HTML and XHTML  
9th Edition

## Tutorial 10

### Programming with JavaScript



New Perspectives on HTML and XHTML, Comprehensive

## Objectives

- Learn the history of JavaScript
- Create a script element
- Understand basic JavaScript syntax
- Write text to a Web page with JavaScript
- Learn about JavaScript data types

New Perspectives on HTML and XHTML, Comprehensive

## Objectives

- Declare and work with variables
- Create and call a JavaScript function
- Access an external JavaScript file
- Add comments to JavaScript code
- Learn about basic debugging techniques and tools

New Perspectives on HTML and XHTML, Comprehensive

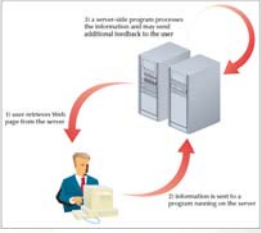
## Introducing JavaScript

- Server-side programs are placed on the server that hosts a Web site
  - Can be problematic
- Client-side programming runs programs on each user's computer

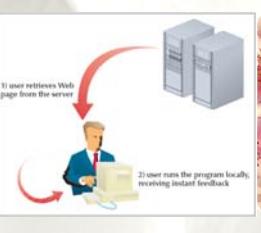
New Perspectives on HTML and XHTML, Comprehensive

## Introducing JavaScript

### Server-Side Programming



### Client-Side Programming



New Perspectives on HTML and XHTML, Comprehensive

## The Development of JavaScript

- JavaScript is a subset of Java
- Differences between Java and JavaScript:
  - Java is a compiled language
  - JavaScript is an interpreted language

New Perspectives on HTML and XHTML, Comprehensive

## Comparing Java and JavaScript

Java	JavaScript
A compiled language	An interpreted language
Requires the JDK (Java Development Kit) to create the applet	Requires a text editor
Requires a Java virtual machine or interpreter to run the applet	Requires a browser that can interpret JavaScript code
Applet files are distinct from the HTML and XHTML code	JavaScript programs are integrated with HTML and XHTML code
Source code is hidden from the user	Source code is accessible to the user
Powerful, requiring programming knowledge and experience	Simpler, requiring less programming knowledge and experience
Secure; programs cannot write content to the hard disk	Secure; programs cannot write content to the hard disk; however, there are more security holes than in Java
Programs run on the client side	Programs run on the client side

New Perspectives on HTML and XHTML, Comprehensive

7

## The Development of JavaScript

- Jscript is a version of JavaScript supported by Internet Explorer
- The European Computer Manufacturers Association (ECMA) develops scripting standards
  - The standard is called ECMAScript but browsers still generally call it JavaScript

New Perspectives on HTML and XHTML, Comprehensive

8

## Working with the Script Element

- A JavaScript program can either be placed directly in a Web page file or saved in an external text file
- Insert a client-side script in a Web page when using the script element

```
<script type="mime-type">  
    script commands  
</script>
```

New Perspectives on HTML and XHTML, Comprehensive

9

## Inserting JavaScript into a Web Page File

- Each statement—also known as a command—is a single line that indicates an action for the browser to take
- The semicolon notifies the browser that it has reached the end of the statement

New Perspectives on HTML and XHTML, Comprehensive

10

## Writing Output to the Web Page

- An object is any item—from the browser window itself to a document displayed in the browser to an element displayed within the document
- A method is a process by which JavaScript manipulates or acts upon the properties of an object

New Perspectives on HTML and XHTML, Comprehensive

11

## Writing Output to the Web Page

- To write text to a Web page, use the following JavaScript commands:

```
document.write("text");  
  
or  
document.writeln("text");
```

where *text* is the content to be written to the page. The `document.write()` and `document.writeln()` methods are identical, except that the `document.writeln()` method preserves any line breaks in the text string

New Perspectives on HTML and XHTML, Comprehensive

12

## Understanding JavaScript Syntax

- JavaScript is case sensitive
- Ignores most occurrences of extra white space
- Do not break a statement into several lines
- The + symbol used in this command combines several text strings into a single text string

New Perspectives on HTML and XHTML, Comprehensive

13

## Working with Variables

- A variable is a named item in a program that stores information
- Most JavaScript programs use variables to represent values and text strings

New Perspectives on HTML and XHTML, Comprehensive

14

## Declaring a JavaScript Variable

- You can declare variables with any of the following JavaScript commands:

```
var variable;  
var variable = value;  
variable = value;
```

where *variable* is the name of the variable and *value* is the initial value of the variable. The first command creates the variable without assigning it a value; the second and third commands both create the variable and assign it a value

New Perspectives on HTML and XHTML, Comprehensive

15

## Working with Variables and Data

- JavaScript variable types:
  - Numeric variables
  - String variables
  - Boolean variables
  - Null variables
- You must **declare** a variable before using it

New Perspectives on HTML and XHTML, Comprehensive

16

## Working with Variables and Data

- Numeric variable- any number, such as 13, 22.5, etc.
  - Can also be expressed in scientific notation
- String variable- any group of text characters, such as "Hello" or "Happy Holidays!"
  - Must be enclosed within either double or single quotations (but not both)
- Boolean variable- accepts only true and false values
- Null variable- has no value at all

New Perspectives on HTML and XHTML, Comprehensive

17

## Working with Variables and Data

- JavaScript is a weakly typed language
- The + symbol can be used with either numeric values or text strings

```
var total = 5 + 4;
```

```
var emLink = "cadler" + "@" +  
"mpl.gov";
```

New Perspectives on HTML and XHTML, Comprehensive

18

## Creating a JavaScript Function

- A function is a collection of commands that performs an action or returns a value
- A function name identifies a function
- Parameters are values used by the function
- The function is executed only when called by another JavaScript command  
*function\_name(parameter values)*

## Creating a JavaScript Function

```
<title>Monroe Public Library</title>
<link href="mp1styles.css" rel="stylesheet" type="text/css" />
<script type="text/javascript">
  function showEM(username, email) {
    var email = username + "@ " + email;
    document.write("<a href='mailto:' + email + '>");
    document.write(email);
  }
</script>
</head>
```

## Creating a Function to Return a Value

- For a function to return a value, it must include a return statement

```
function function_name(parameters) {
  JavaScript commands
  return value;
}
```

## Accessing an External JavaScript File

- The code to access an external script file is:

```
<script src="url" type="mime-type"></script>
```

- Place all script elements that reference external files in the document head

## Accessing an External JavaScript File

```
function stringReverse(str) {
  if (!str) return "";
  var strArray = str.split("");
  strArray.reverse();
  return strArray.join("");
}

<title>Monroe Public Library</title>
<link href="mp1styles.css" rel="stylesheet" type="text/css" />
<script src="js/rev.js" type="text/javascript"></script>
<script type="text/javascript">
  function showEM(username, email) {
    var email = username + "@ " + email;
    document.write("<a href='mailto:' + email + '>");
    document.write(email);
  }
</script>
</head>
```

## Commenting JavaScript Code

- Commenting your code is an important programming practice

```
// comment text
```

```
<script type="text/javascript">
  function showEM(username, email) {
    // the show() function displays a link in the user's e-mail address
    // the text of the user and e-mail address are entered in
    // reverse order to show e-mail harvestors.
    username = stringReverse(username); // reverse the text of the username parameter
    observer = stringReverse(email); // reverse the text of the observer parameter
    var email = username + "@ " + observer; // combine the text of username and observer
    document.write("<a href='mailto:' + email + '>");
    document.write(email);
  }
</script>
```

