HTML, XHTML, and XML

3rd Edition

Tutorial 10

Programming with JavaScript

http://www.w3schools.com/js

Thanks to the author of the textbook for providing these slides. I made slight changes/additions.
Turgay Korkmaz

Objectives

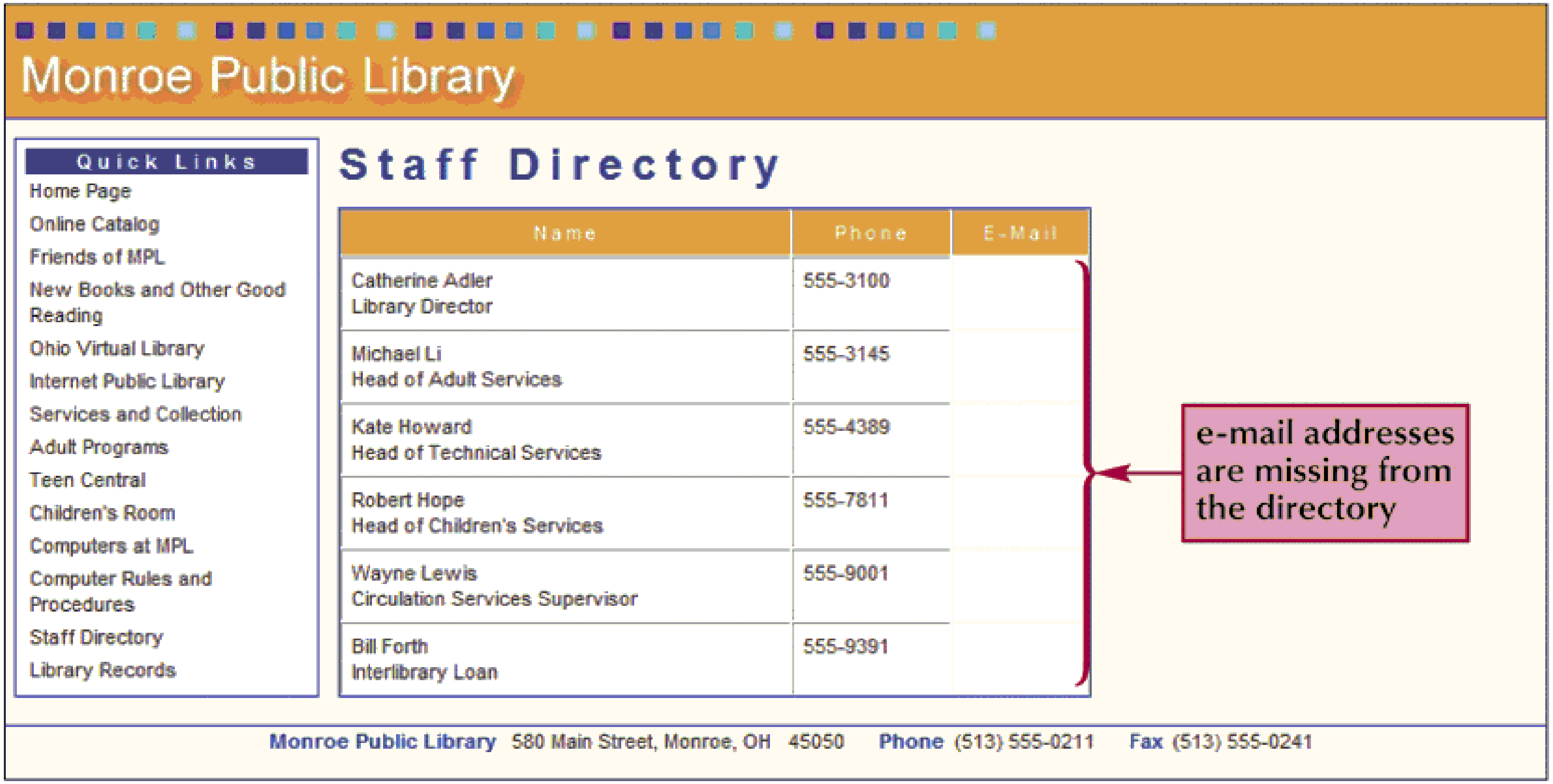
- Learn the history of JavaScript
- Create a script element
- Understand basic JavaScript syntax
- Write text to a Web page with JavaScript
- Learn about JavaScript data types

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

Tutorial_10/tutorial

Textbook
Pg 564



Monroe Public Library

Quick Links

- Home Page
- Online Catalog
- Friends of MPL
- New Books and Other Good Reading
- Ohio Virtual Library
- Internet Public Library
- Services and Collection
- Adult Programs
- Teen Central
- Children's Room
- Computers at MPL
- Computer Rules and Procedures
- Staff Directory
- Library Records

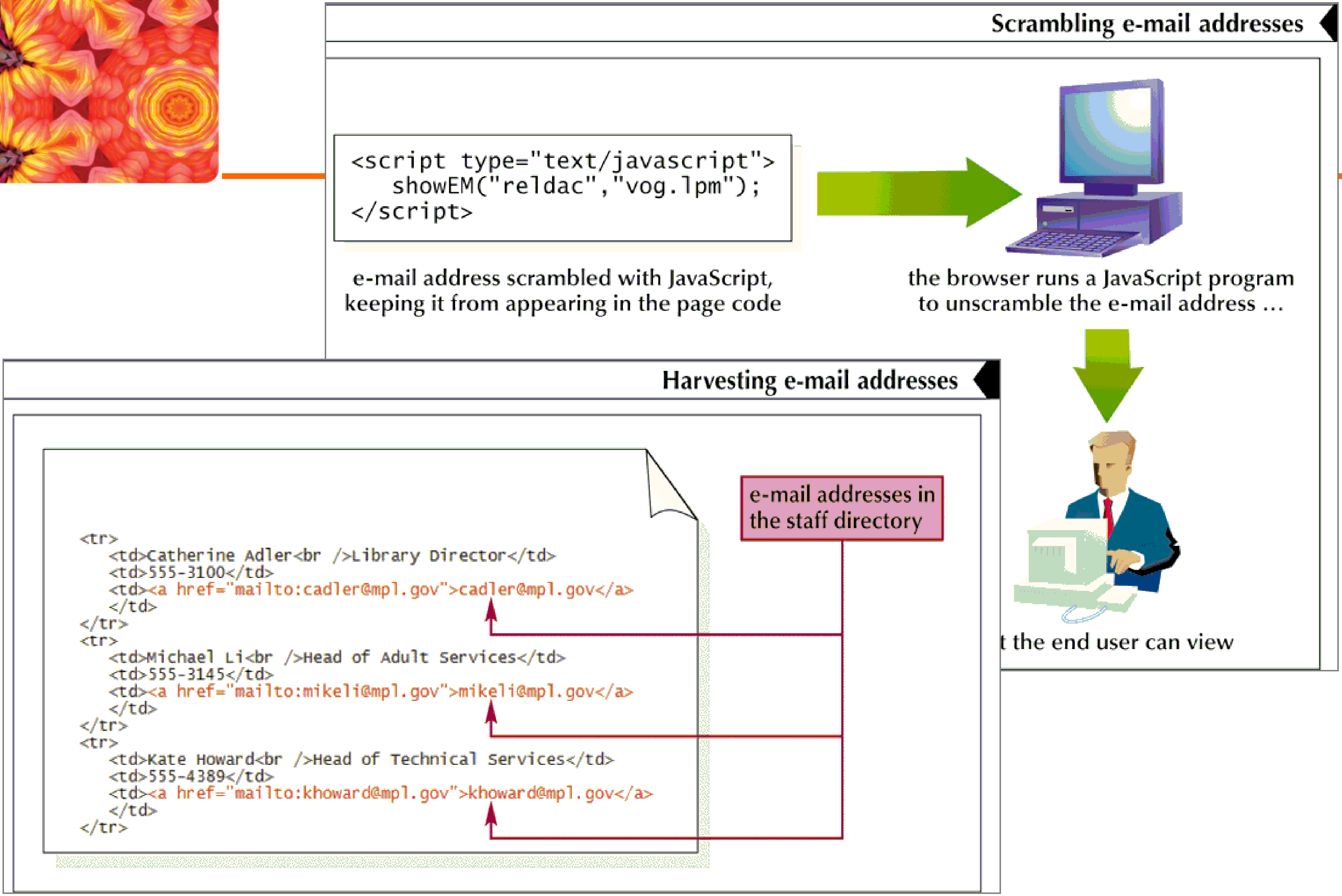
Staff Directory

Name	Phone	E-Mail
Catherine Adler Library Director	555-3100	
Michael Li Head of Adult Services	555-3145	
Kate Howard Head of Technical Services	555-4389	
Robert Hope Head of Children's Services	555-7811	
Wayne Lewis Circulation Services Supervisor	555-9001	
Bill Forth Interlibrary Loan	555-9391	

e-mail addresses are missing from the directory

Monroe Public Library 580 Main Street, Monroe, OH 45050 Phone (513) 555-0211 Fax (513) 555-0241

Scramble the e-mail addresses within the HTML code, but they should be viewable when the page is rendered by a browser…. *Protection against E-mail harvester – spammer…*



Scrambling e-mail addresses

```
<script type="text/javascript">
   showEM("reldac","vog.lpm");
</script>
```

e-mail address scrambled with JavaScript, keeping it from appearing in the page code

the browser runs a JavaScript program to unscramble the e-mail address …

t the end user can view

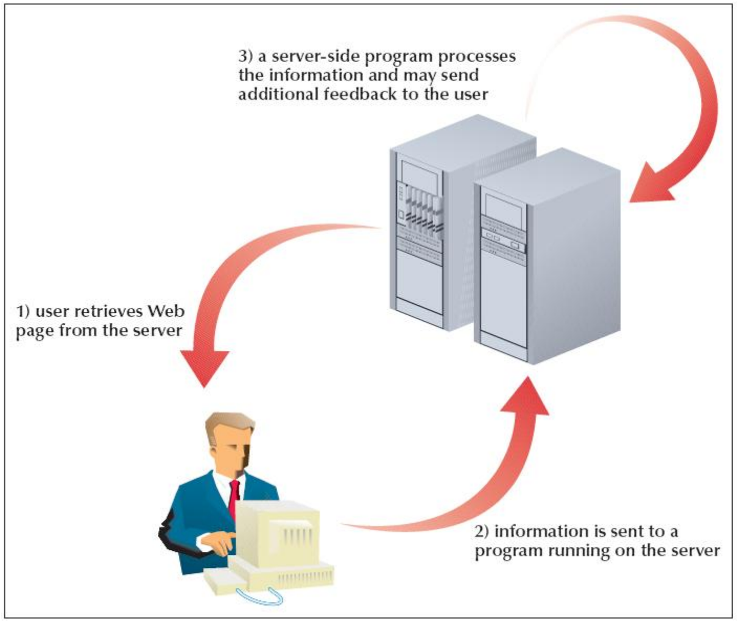
Harvesting e-mail addresses

```
<tr>
   <td>Catherine Adler<br />Library Director</td>
   <td>555-3100</td>
   <td><a href="mailto:cadler@mpl.gov">cadler@mpl.gov</a>
   </td>
</tr>
<tr>
   <td>Michael Li<br />Head of Adult Services</td>
   <td>555-3145</td>
   <td><a href="mailto:mikeli@mpl.gov">mikeli@mpl.gov</a>
   </td>
</tr>
<tr>
   <td>Kate Howard<br />Head of Technical Services</td>
   <td>555-4389</td>
   <td><a href="mailto:khoward@mpl.gov">khoward@mpl.gov</a>
   </td>
</tr>
```

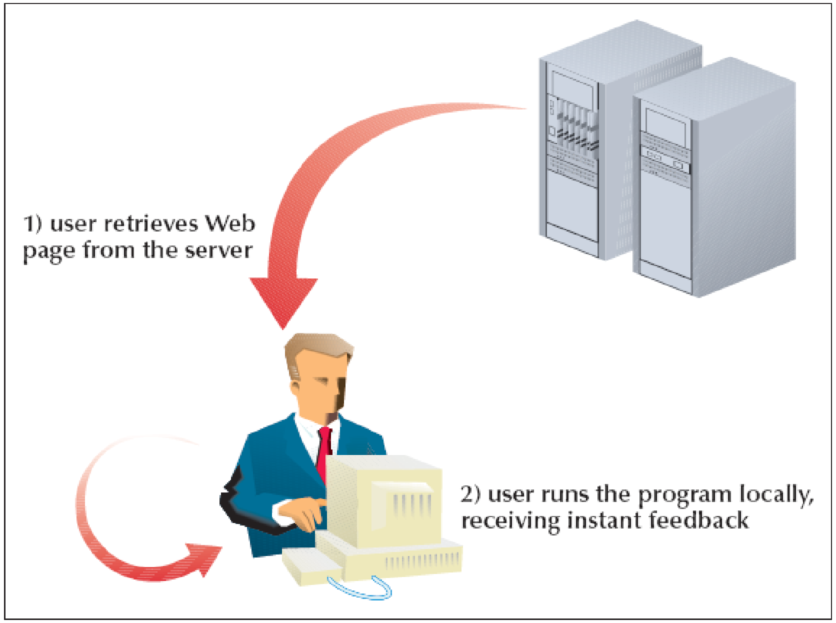
e-mail addresses in the staff directory

Introducing JavaScript

- **Server-side** programs are placed on the server that hosts a Web site


3) a server-side program processes the information and may send additional feedback to the user
1) user retrieves Web page from the server
2) information is sent to a program running on the server

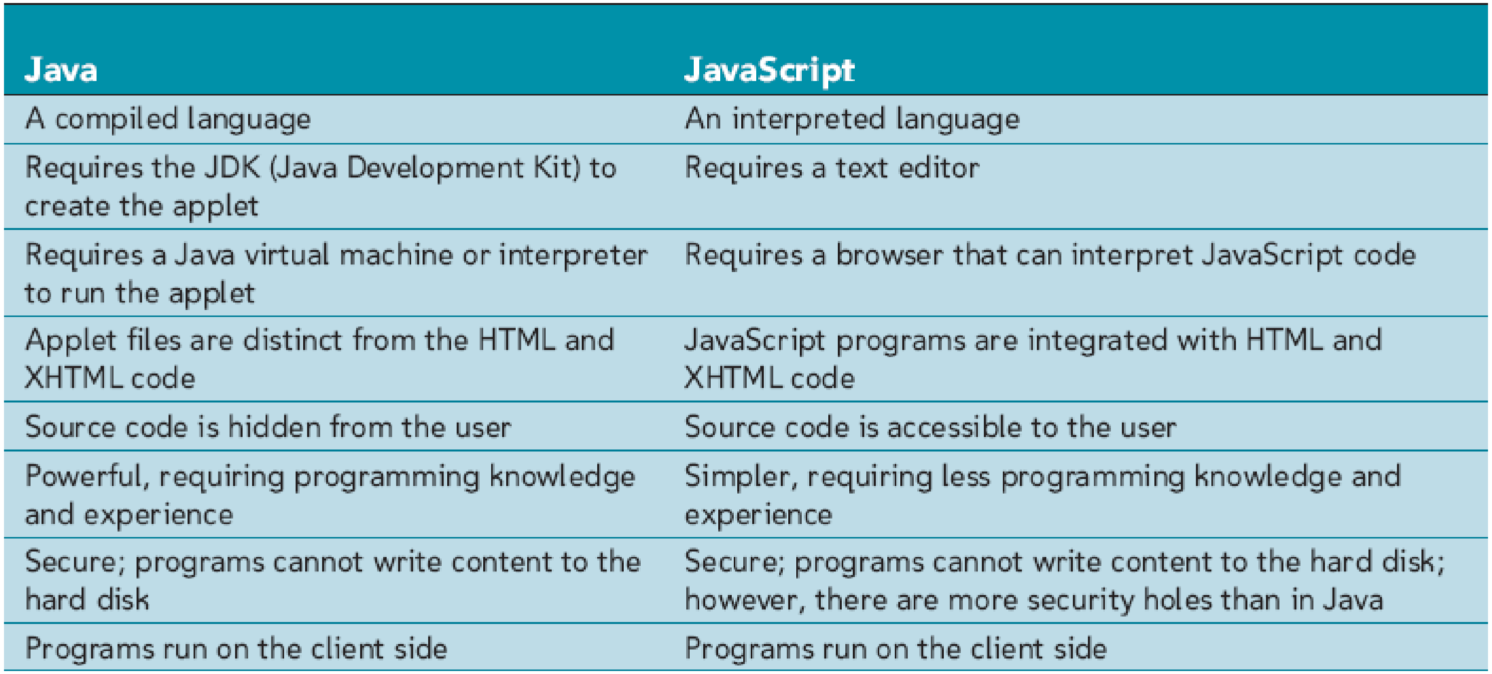
- **Client-side** programming runs programs on each user's computer


1) user retrieves Web page from the server
2) user runs the program locally, receiving instant feedback

In practice both are combined.

The Development of JavaScript

- JavaScript is a subset of Java, but different in many ways



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

- Not as powerful as Java, but simple to use and meets the needs of most users

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 is JavaScript

WORKING WITH THE SCRIPT ELEMENT

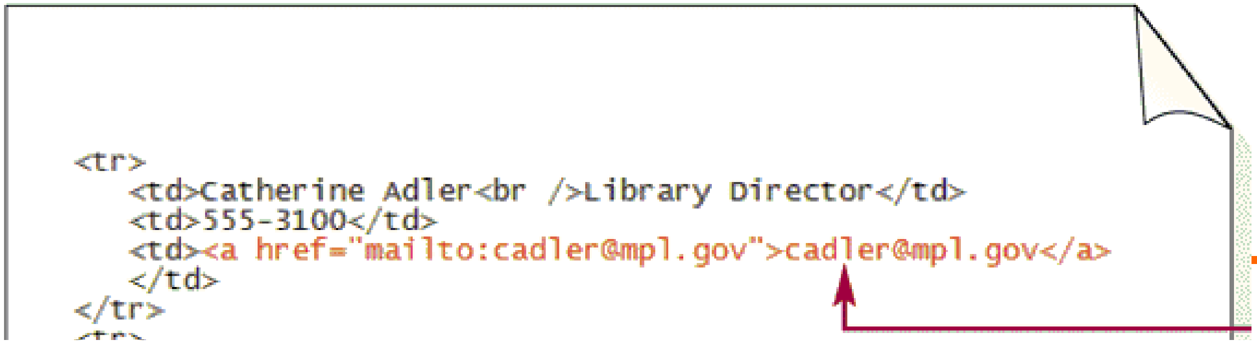
Creating Script Element

- 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 by using the **script** element

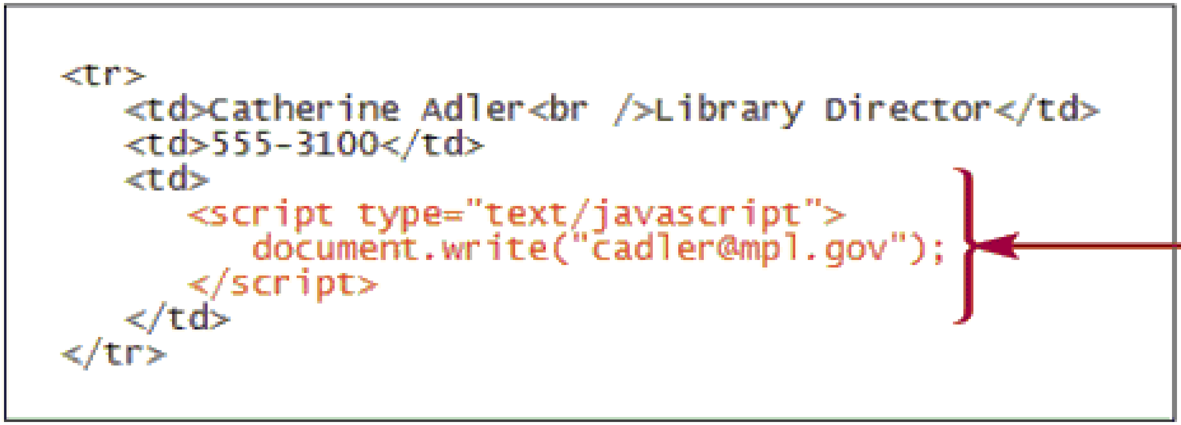
```
<script type="mime-type”>
   script commands
</script>

<script type=“text/JavaScript”>
   JavaScript commands
</script>
```

Inserting JavaScript into a Web Page File

- When a web browser encounters a script element within an HTML file, the browser treats any line (statement) in the script element as commands to be run
  - 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
- Script elements are processed in the order they appear in the HTML file. The can be in the head or body section.
- If they are in body, the browser runs them as it loads the different elements of the Web page
- JavaScript commands created in one script element can be referenced by commands in another script element

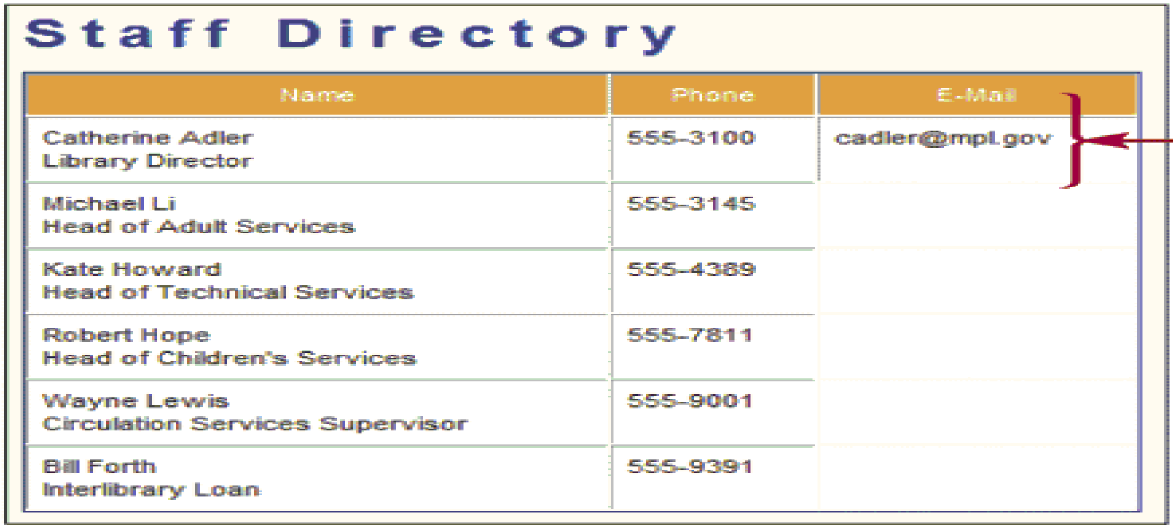


```
<tr>
   <td>Catherine Adler<br />Library Director</td>
   <td>555-3100</td>
   <td><a href="mailto:cadler@mpl.gov">cadler@mpl.gov</a>
   </td>
</tr>
```



```
<tr>
   <td>Catherine Adler<br />Library Director</td>
   <td>555-3100</td>
   <td>
      <script type="text/javascript">
         document.write("cadler@mpl.gov");
      </script>
   </td>
</tr>
```

script to write content to the Web document



Staff Directory

Name	Phone	E-Mail
Catherine Adler Library Director	555-3100	cadler@mpl.gov
Michael Li Head of Adult Services	555-3145	
Kate Howard Head of Technical Services	555-4389	
Robert Hope Head of Children's Services	555-7811	
Wayne Lewis Circulation Services Supervisor	555-9001	
Bill Forth Interlibrary Loan	555-9391	

e-mail address inserted using JavaScript

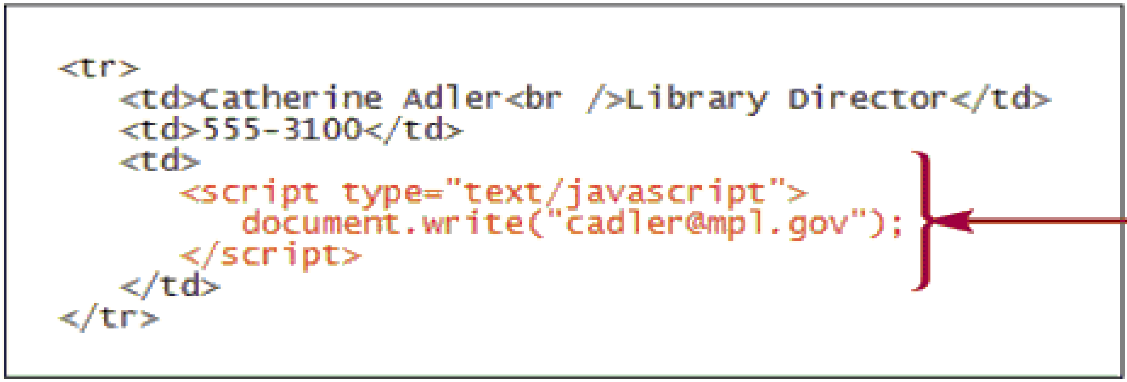
Objects and Methods

- In JavaScript, many commands involve working with objects in the **Web page** and **browser**
- An **`object`** is any item—from the browser window itself to a document displayed in the browser to an element displayed within the document (mouse pointer, scrollbar etc.)
- A **`method`** is a process by which JavaScript manipulates or acts upon the properties of an object

```
<script type=“text/JavaScript”>
   document.write(“some text”);
</script>
```

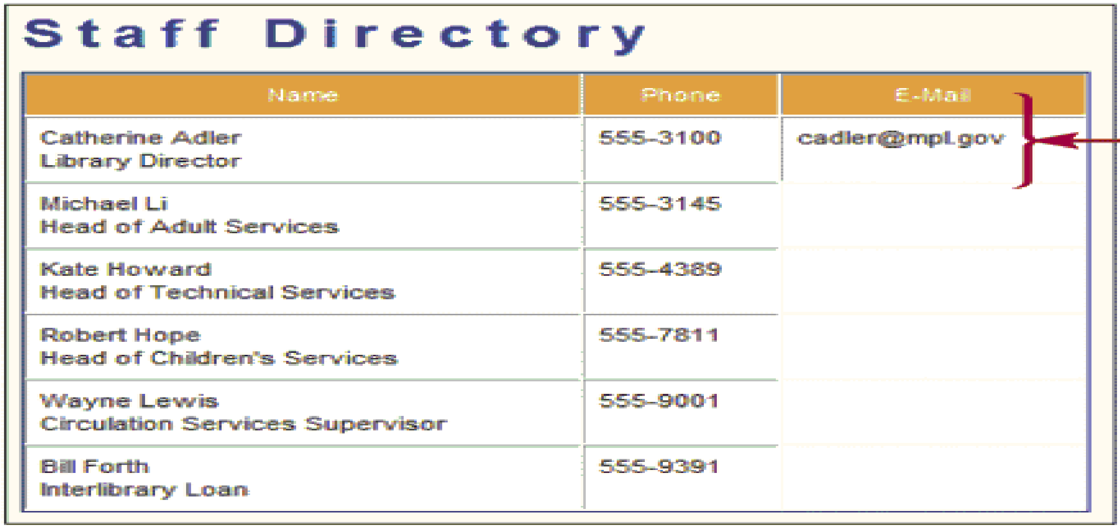
Writing Output to the Web Page

Textbook
Pg 572



```
<tr>
   <td>Catherine Adler<br />Library Director</td>
   <td>555-3100</td>
   <td>
      <script type="text/javascript">
         document.write("cadler@mpl.gov");
      </script>
   </td>
</tr>
```

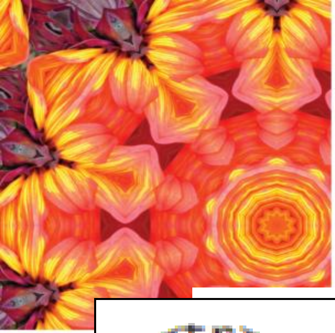
script to write content to the Web document



Staff Directory

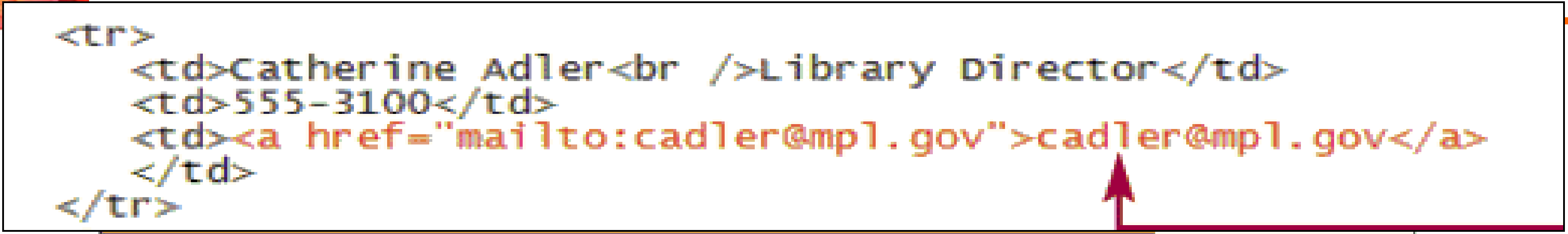
Name	Phone	E-Mail
Catherine Adler Library Director	555-3100	cadler@mpl.gov
Michael Li Head of Adult Services	555-3145	
Kate Howard Head of Technical Services	555-4389	
Robert Hope Head of Children's Services	555-7811	
Wayne Lewis Circulation Services Supervisor	555-9001	
Bill Forth Interlibrary Loan	555-9391	

e-mail address inserted using JavaScript

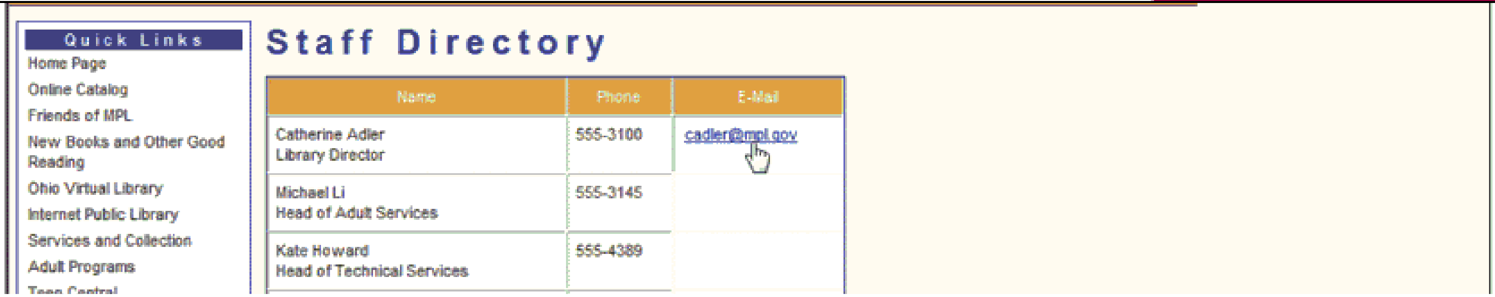


Writing Output to the Web Page

Textbook
Pg 574



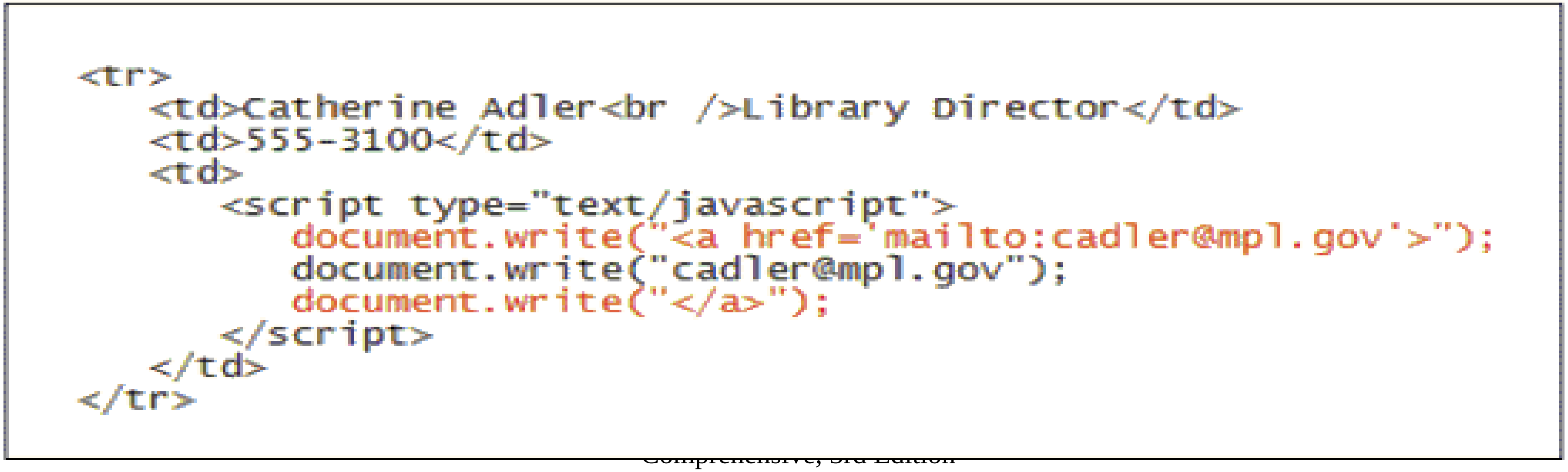
```
<tr>
   <td>Catherine Adler<br />Library Director</td>
   <td>555-3100</td>
   <td><a href="mailto:cadler@mpl.gov">cadler@mpl.gov</a>
   </td>
</tr>
```



Quick Links
Home Page
Online Catalog
Friends of MPL
New Books and Other Good Reading
Ohio Virtual Library
Internet Public Library
Services and Collection
Adult Programs
Teen Central

Staff Directory

Name	Phone	E-Mail
Catherine Adler Library Director	555-3100	cadler@mpl.gov
Michael Li Head of Adult Services	555-3145	
Kate Howard Head of Technical Services	555-4389	



```
<tr>
   <td>Catherine Adler<br />Library Director</td>
   <td>555-3100</td>
   <td>
      <script type="text/javascript">
         document.write("<a href='mailto:cadler@mpl.gov'>");
         document.write("cadler@mpl.gov");
         document.write("</a>");
      </script>
   </td>
</tr>
```

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”);
```

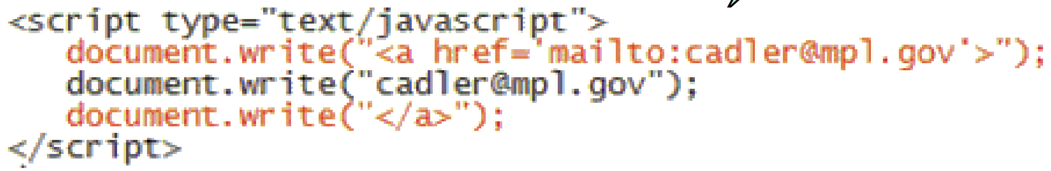
where *text* is the content to be written to the page. The doucment.write() and document.writeln() methods are identical, except that the document.writeln() method preserves any line breaks in the text string

Understanding JavaScript Syntax

- JavaScript is case sensitive: doc vs. Doc vs dOc
- Ignores most occurrences of extra white space
- Do not break a statement into several lines
- The + symbol used in the following command combines several text strings into a single text string

```
<script type=“text/JavaScript”>
    document.write(“some text” +
“ Some other text” +
“more text”);
</script>
```

Not good



```
<script type="text/javascript">
   document.write("<a href='mailto:cadler@mpl.gov'>");
   document.write("cadler@mpl.gov");
   document.write("</a>");
</script>
```

XHTML my try to process <>& in JavaScript, to avoid this problem

```
<script type=“text/JavaScript”>
<![CDATA[
  JavaScript commands
]]>
</script>
```

SESSION 10.2
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

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
- You must **declare** a variable before using it

Working with Variables and Data

- JavaScript variable types:
  - Numeric variables
    - any number, such as 2, 3.4 etc `year = 2011;`
    - rather than `year=“2011”;`
  - String variables
    - any group of text characters, `name = “turgay”;`
    - Must be enclosed within either double or single quotations (but not both)
  - Boolean variables
    - accepts only true and false values `useIE = false;`
  - Null variables
    - has no value at all `link = null;`

Working with Variables and Data

- Variable’s type is determined by the context. So variables can switch type

```
Month = 5;
Month = “March”;
```

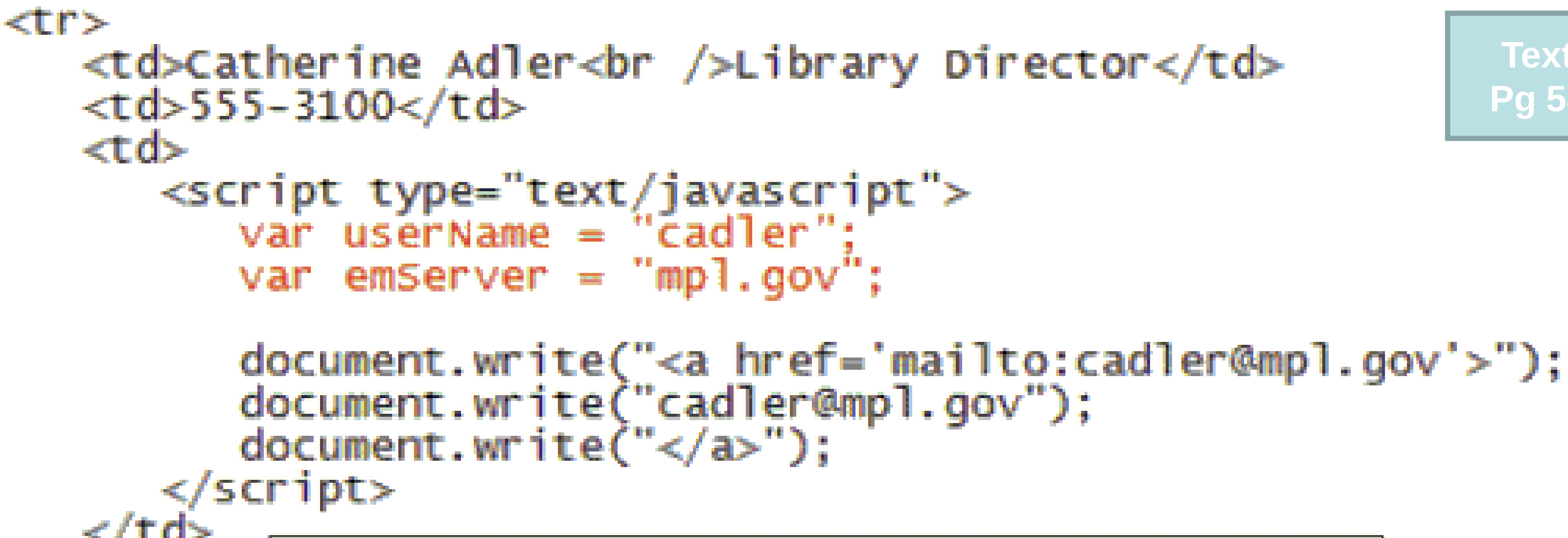
- JavaScript is a weakly typed language
- The + symbol can be used with either numeric values or text strings

```
var total = 5 + 4;
var emLink = "cadler" + "@" + "mpl.gov";
```

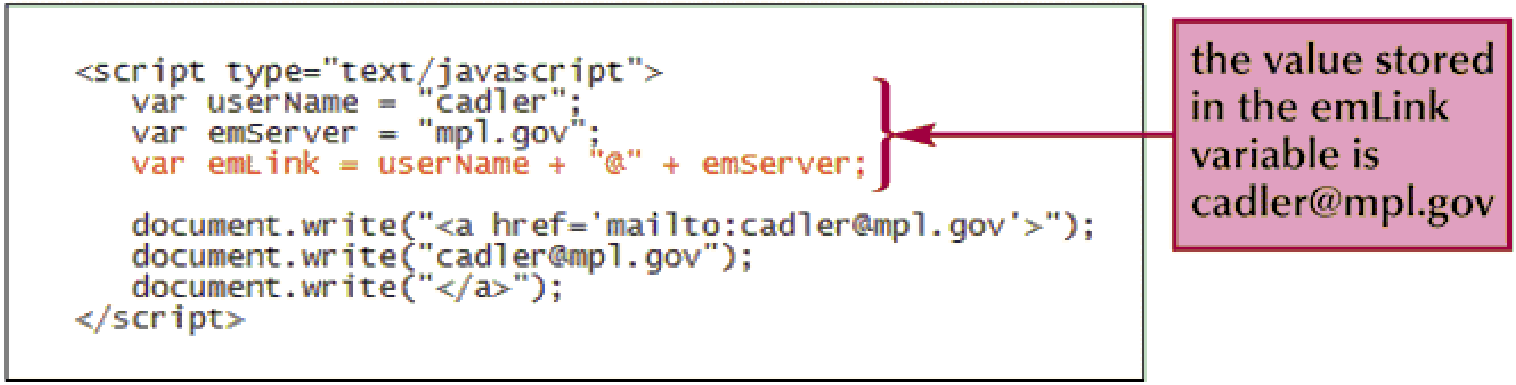
- How about?

```
x=5; y=“4”; z=x+y;          text string 54, why?
```

Textbook
Pg 580-2-3

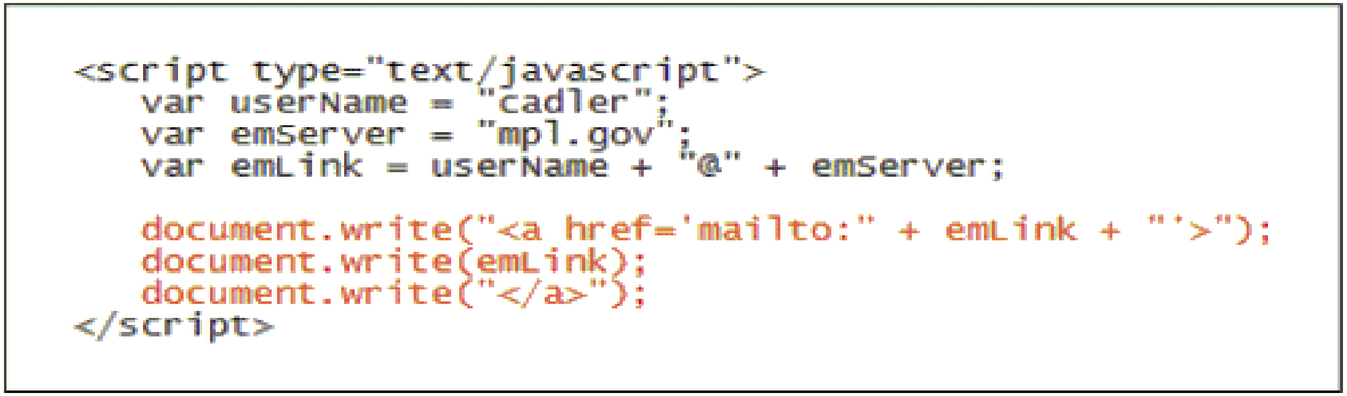


```
<tr>
   <td>Catherine Adler<br />Library Director</td>
   <td>555-3100</td>
   <td>
      <script type="text/javascript">
         var userName = "cadler";
         var emServer = "mpl.gov";

         document.write("<a href='mailto:cadler@mpl.gov'>");
         document.write("cadler@mpl.gov");
         document.write("</a>");
      </script>
   </td>
</tr>
```



```
<script type="text/javascript">
   var userName = "cadler";
   var emServer = "mpl.gov";
   var emLink = userName + "@" + emServer;

   document.write("<a href='mailto:cadler@mpl.gov'>");
   document.write("cadler@mpl.gov");
   document.write("</a>");
</script>
```

the value stored in the emLink variable is cadler@mpl.gov



```
<script type="text/javascript">
   var userName = "cadler";
   var emServer = "mpl.gov";
   var emLink = userName + "@" + emServer;

   document.write("<a href='mailto:" + emLink + "'>");
   document.write(emLink);
   document.write("</a>");
</script>
```

Will you use the previous code for each individual listed in the staff directory?

When you need to reuse the same commands through your web page, you store the commands in a function!

JAVASCRIPT FUNCTIONS

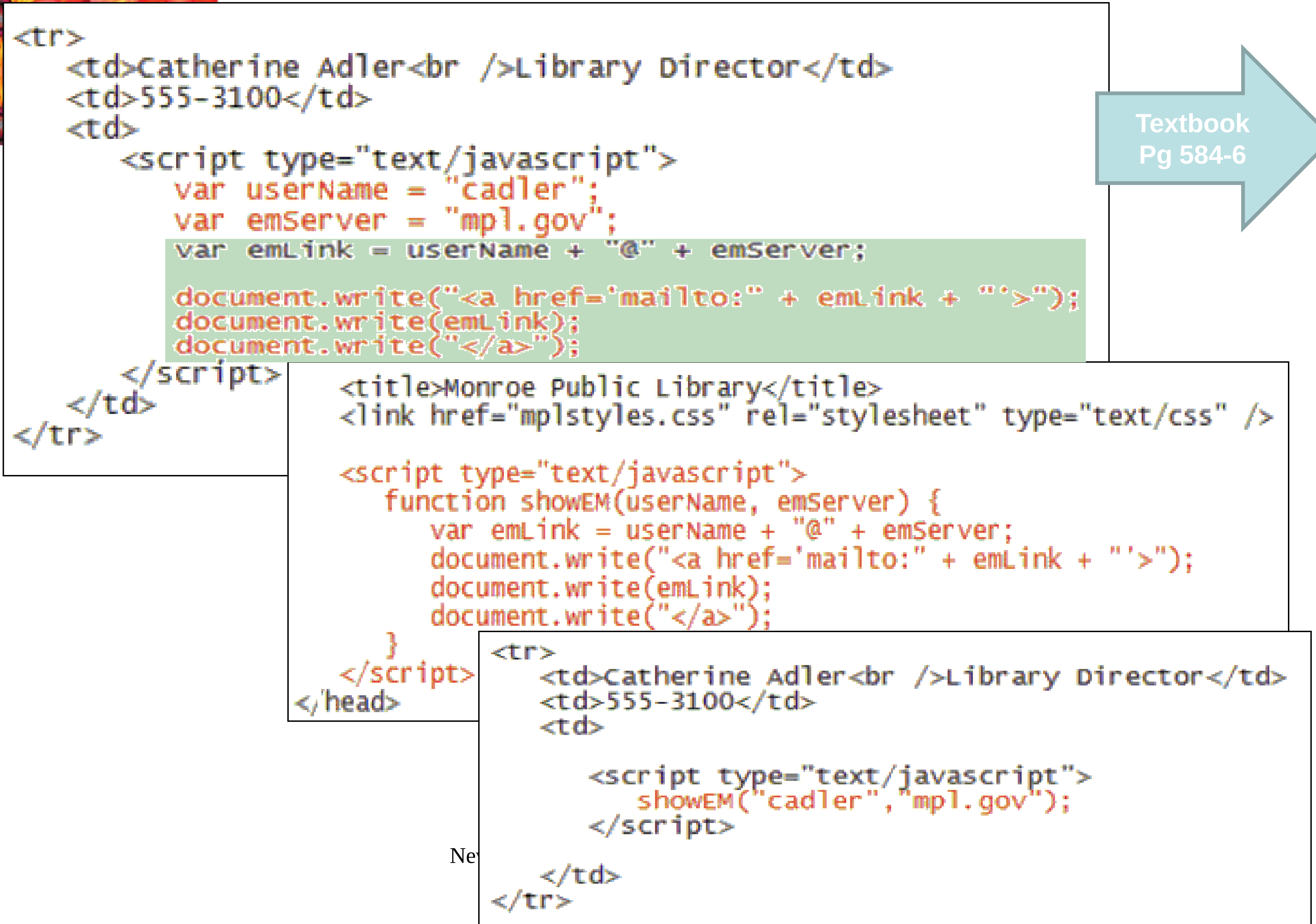
Creating a JavaScript Function

- A function is a collection of commands
  - It performs an action and/or
  - It returns a value

```
function function_name(parameter values){
    JavaScript commands
}
```

- A function name identifies a function
- Parameters are values used by the function
- The function is executed only when **called** by another JavaScript command
- Put all functions in the **head** section

JavaScript Function Performing an Action

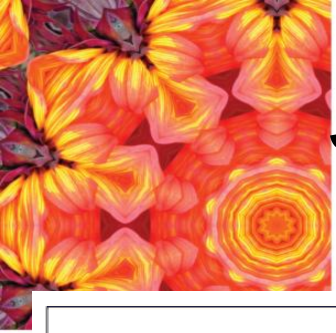


```
<tr>
   <td>Catherine Adler<br />Library Director</td>
   <td>555-3100</td>
   <td>
      <script type="text/javascript">
         var userName = "cadler";
         var emServer = "mpl.gov";
         var emLink = userName + "@" + emServer;

         document.write("<a href='mailto:" + emLink + "'>");
         document.write(emLink);
         document.write("</a>");
      </script>
   </td>
</tr>
```

```
   <title>Monroe Public Library</title>
   <link href="mplstyles.css" rel="stylesheet" type="text/css" />

   <script type="text/javascript">
      function showEM(userName, emServer) {
         var emLink = userName + "@" + emServer;
         document.write("<a href='mailto:" + emLink + "'>");
         document.write(emLink);
         document.write("</a>");
      }
   </script>
</head>
```

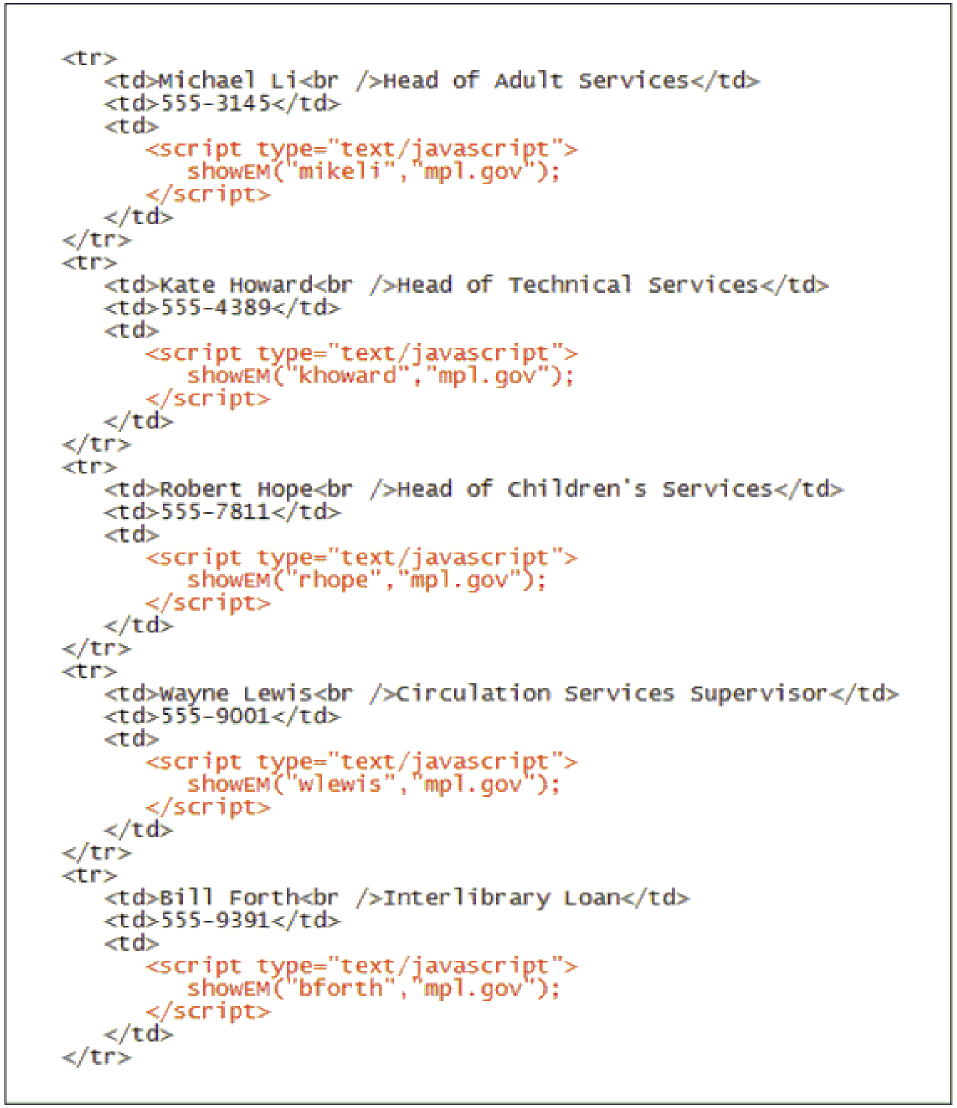
```
<tr>
   <td>Catherine Adler<br />Library Director</td>
   <td>555-3100</td>
   <td>

      <script type="text/javascript">
         showEM("cadler","mpl.gov");
      </script>

   </td>
</tr>
```

Textbook
Pg 584-6

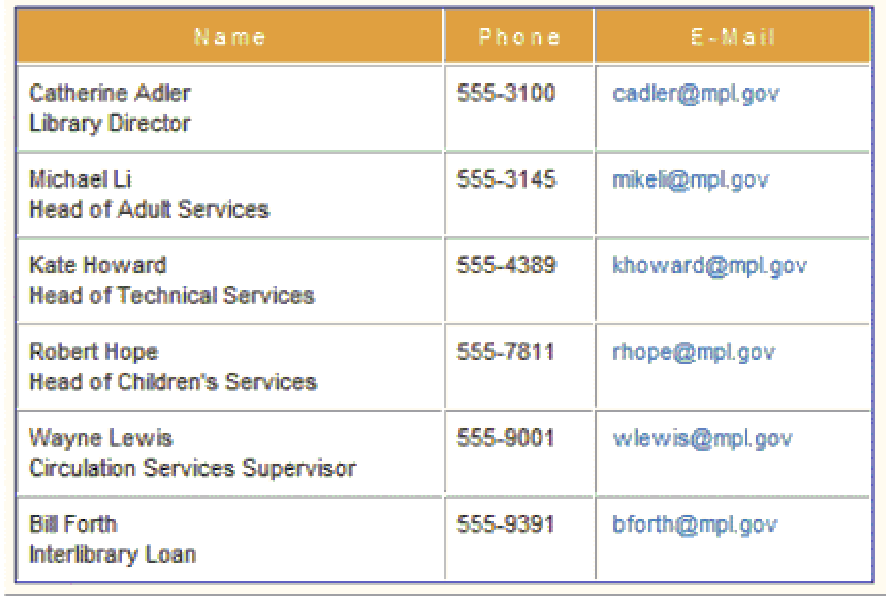


JavaScript Function Performing an Action

Textbook
Pg 587



```
<tr>
   <td>Michael Li<br />Head of Adult Services</td>
   <td>555-3145</td>
   <td>
      <script type="text/javascript">
         showEM("mikeli","mpl.gov");
      </script>
   </td>
</tr>
<tr>
   <td>Kate Howard<br />Head of Technical Services</td>
   <td>555-4389</td>
   <td>
      <script type="text/javascript">
         showEM("khoward","mpl.gov");
      </script>
   </td>
</tr>
<tr>
   <td>Robert Hope<br />Head of Children's Services</td>
   <td>555-7811</td>
   <td>
      <script type="text/javascript">
         showEM("rhope","mpl.gov");
      </script>
   </td>
</tr>
<tr>
   <td>Wayne Lewis<br />Circulation Services Supervisor</td>
   <td>555-9001</td>
   <td>
      <script type="text/javascript">
         showEM("wlewis","mpl.gov");
      </script>
   </td>
</tr>
<tr>
   <td>Bill Forth<br />Interlibrary Loan</td>
   <td>555-9391</td>
   <td>
      <script type="text/javascript">
         showEM("bforth","mpl.gov");
      </script>
   </td>
</tr>
```



Name	Phone	E-Mail
Catherine Adler Library Director	555-3100	cadler@mpl.gov
Michael Li Head of Adult Services	555-3145	mikeli@mpl.gov
Kate Howard Head of Technical Services	555-4389	khoward@mpl.gov
Robert Hope Head of Children's Services	555-7811	rhope@mpl.gov
Wayne Lewis Circulation Services Supervisor	555-9001	wlewis@mpl.gov
Bill Forth Interlibrary Loan	555-9391	bforth@mpl.gov

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;
}
```

```
function CalcArea(length, width){
        var area = length * width;
        return area;
}
```

```
var x = 8;
var y = 6;
var z = CalcArea(x, y);
```

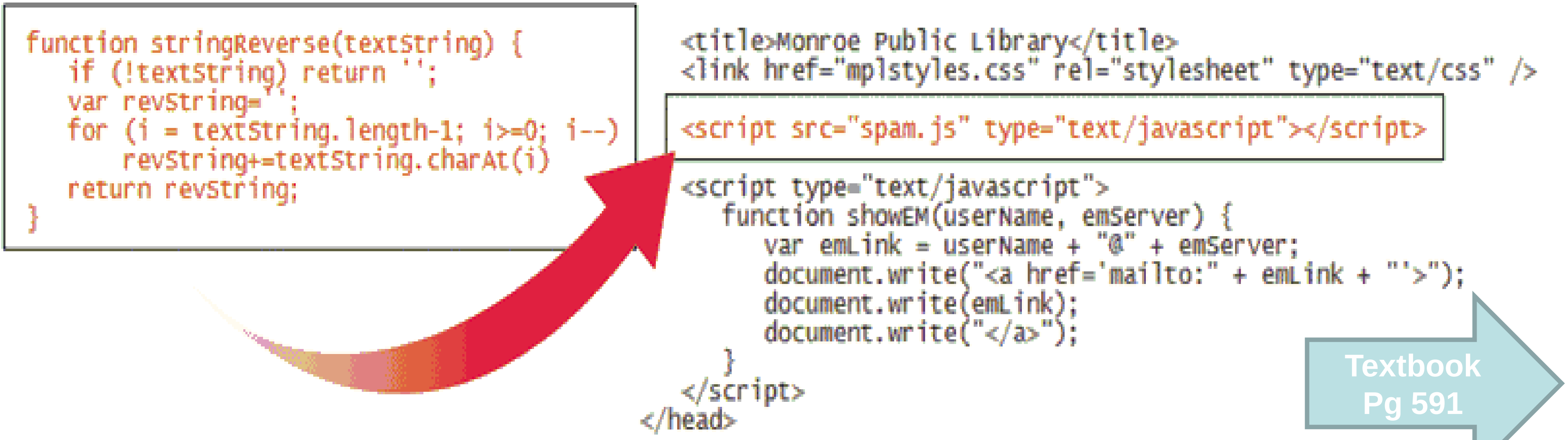
Session 10.3

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

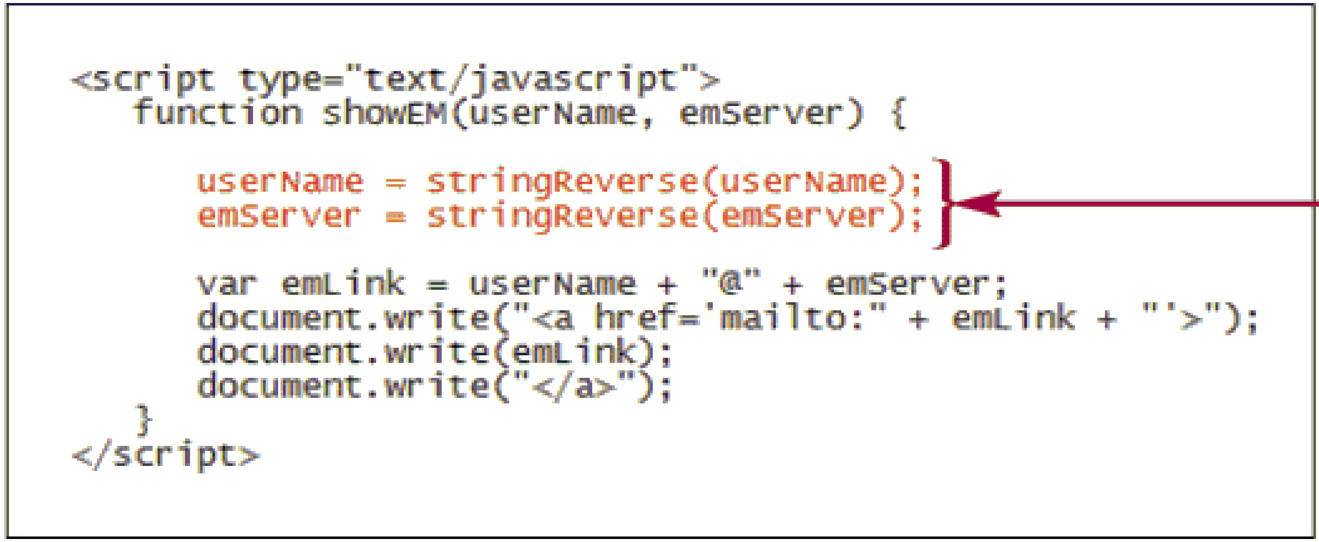


```
function stringReverse(textString) {
   if (!textString) return '';
   var revString='';
   for (i = textString.length-1; i>=0; i--)
      revString+=textString.charAt(i)
   return revString;
}
```

```
<title>Monroe Public Library</title>
<link href="mplstyles.css" rel="stylesheet" type="text/css" />

<script src="spam.js" type="text/javascript"></script>

<script type="text/javascript">
   function showEM(userName, emServer) {
      var emLink = userName + "@" + emServer;
      document.write("<a href='mailto:" + emLink + "'>");
      document.write(emLink);
      document.write("</a>");
   }
</script>
</head>
```

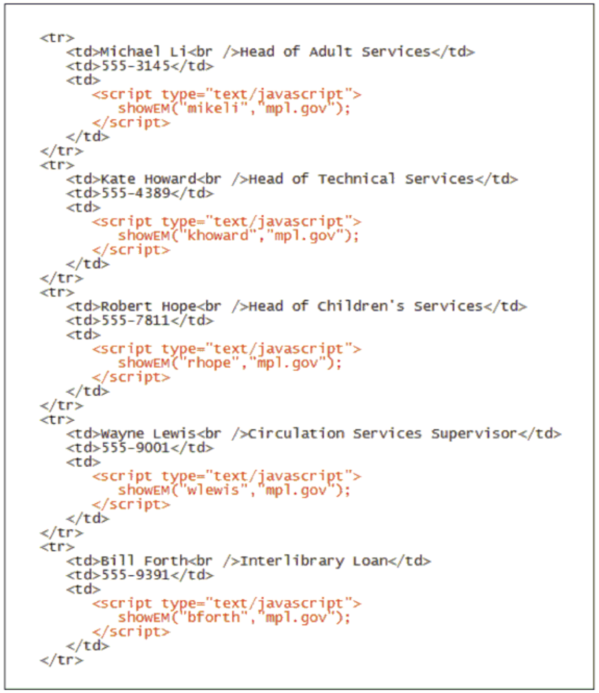
Textbook
Pg 591

Textbook
Pg 592

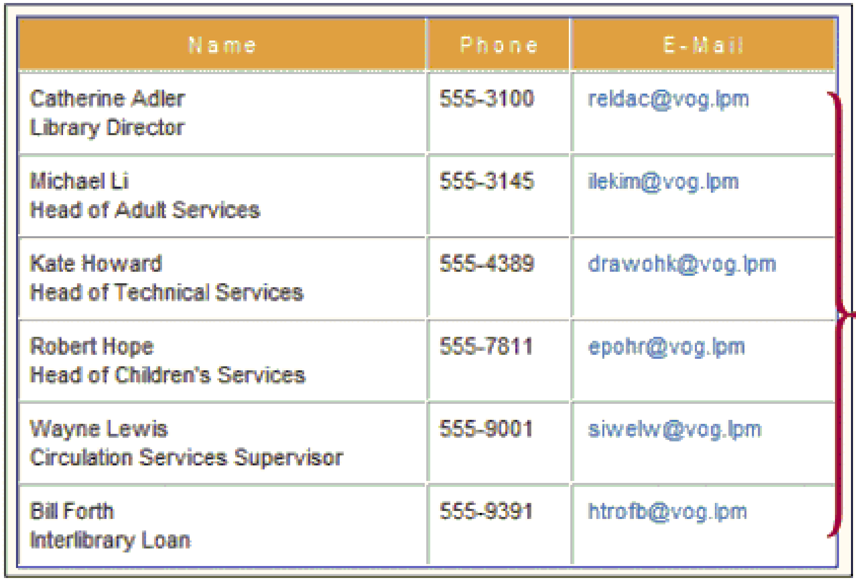


```
<script type="text/javascript">
   function showEM(userName, emServer) {

      userName = stringReverse(userName);
      emServer = stringReverse(emServer);

      var emLink = userName + "@" + emServer;
      document.write("<a href='mailto:" + emLink + "'>");
      document.write(emLink);
      document.write("</a>");
   }
</script>
```

reverse the order of the characters in the userName and emServer parameters



```
<tr>
   <td>Michael Li<br />Head of Adult Services</td>
   <td>555-3145</td>
   <td>
      <script type="text/javascript">
         showEM("mikeli","mpl.gov");
      </script>
   </td>
</tr>
<tr>
   <td>Kate Howard<br />Head of Technical Services</td>
   <td>555-4389</td>
   <td>
      <script type="text/javascript">
         showEM("khoward","mpl.gov");
      </script>
   </td>
</tr>
<tr>
   <td>Robert Hope<br />Head of Children's Services</td>
   <td>555-7811</td>
   <td>
      <script type="text/javascript">
         showEM("rhope","mpl.gov");
      </script>
   </td>
</tr>
<tr>
   <td>Wayne Lewis<br />Circulation Services Supervisor</td>
   <td>555-9001</td>
   <td>
      <script type="text/javascript">
         showEM("wlewis","mpl.gov");
      </script>
   </td>
</tr>
<tr>
   <td>Bill Forth<br />Interlibrary Loan</td>
   <td>555-9391</td>
   <td>
      <script type="text/javascript">
         showEM("bforth","mpl.gov");
      </script>
   </td>
</tr>
```

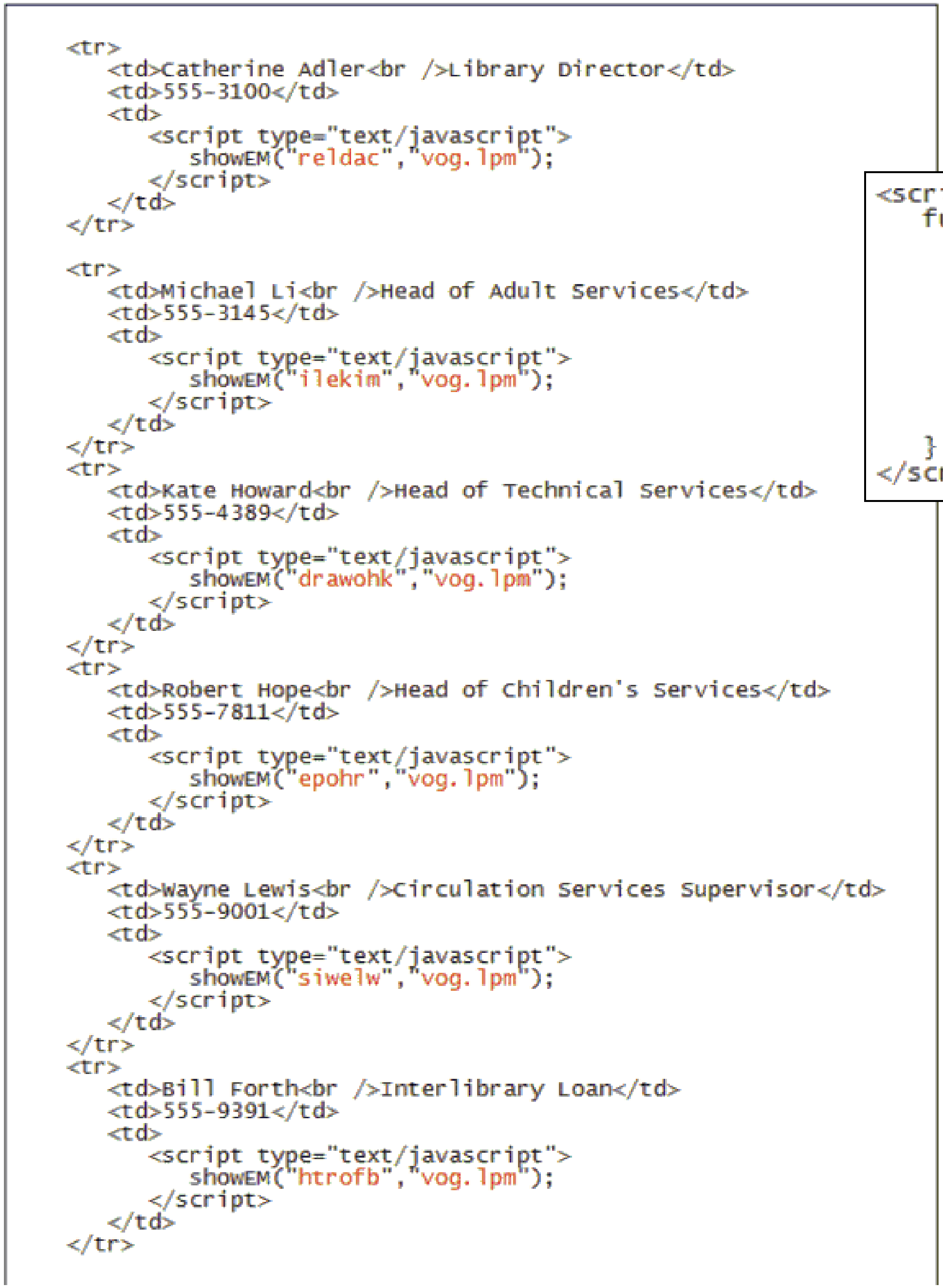


Name	Phone	E-Mail
Catherine Adler Library Director	555-3100	reldac@vog.lpm
Michael Li Head of Adult Services	555-3145	ilekim@vog.lpm
Kate Howard Head of Technical Services	555-4389	drawohk@vog.lpm
Robert Hope Head of Children's Services	555-7811	epohr@vog.lpm
Wayne Lewis Circulation Services Supervisor	555-9001	siwelw@vog.lpm
Bill Forth Interlibrary Loan	555-9391	htrofb@vog.lpm

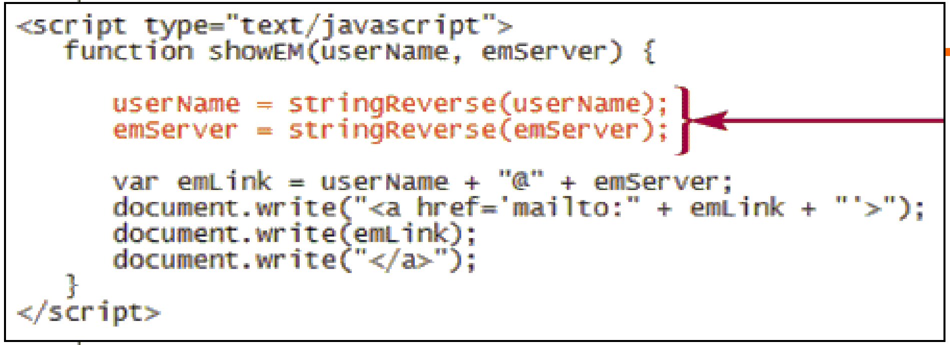
text of each username and e-mail server is reversed

Entering the reversed userName and emServer parameter values

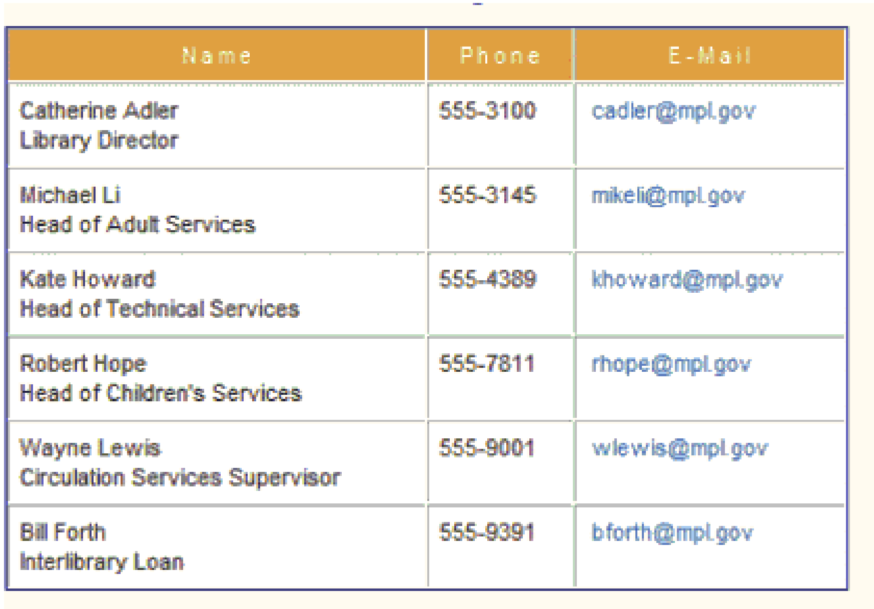
Textbook
Pg 593-4



```
<tr>
   <td>Catherine Adler<br />Library Director</td>
   <td>555-3100</td>
   <td>
      <script type="text/javascript">
         showEM("reldac","vog.lpm");
      </script>
   </td>
</tr>

<tr>
   <td>Michael Li<br />Head of Adult Services</td>
   <td>555-3145</td>
   <td>
      <script type="text/javascript">
         showEM("ilekim","vog.lpm");
      </script>
   </td>
</tr>
<tr>
   <td>Kate Howard<br />Head of Technical Services</td>
   <td>555-4389</td>
   <td>
      <script type="text/javascript">
         showEM("drawohk","vog.lpm");
      </script>
   </td>
</tr>
<tr>
   <td>Robert Hope<br />Head of Children's Services</td>
   <td>555-7811</td>
   <td>
      <script type="text/javascript">
         showEM("epohr","vog.lpm");
      </script>
   </td>
</tr>
<tr>
   <td>Wayne Lewis<br />Circulation Services Supervisor</td>
   <td>555-9001</td>
   <td>
      <script type="text/javascript">
         showEM("siwelw","vog.lpm");
      </script>
   </td>
</tr>
<tr>
   <td>Bill Forth<br />Interlibrary Loan</td>
   <td>555-9391</td>
   <td>
      <script type="text/javascript">
         showEM("htrofb","vog.lpm");
      </script>
   </td>
</tr>
```



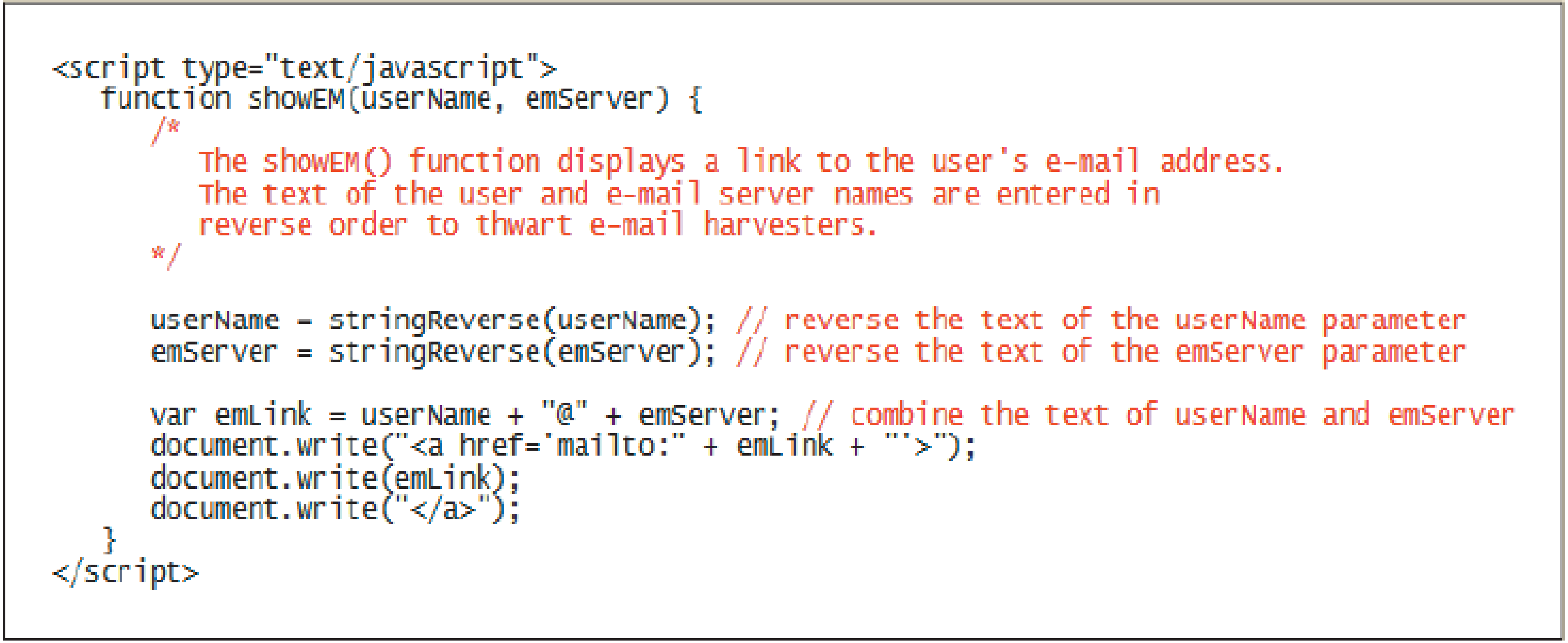
```
<script type="text/javascript">
   function showEM(userName, emServer) {

      userName = stringReverse(userName);
      emServer = stringReverse(emServer);

      var emLink = userName + "@" + emServer;
      document.write("<a href='mailto:" + emLink + "'>");
      document.write(emLink);
      document.write("</a>");
   }
</script>
```



Name	Phone	E-Mail
Catherine Adler Library Director	555-3100	cadler@mpl.gov
Michael Li Head of Adult Services	555-3145	mikeli@mpl.gov
Kate Howard Head of Technical Services	555-4389	khoward@mpl.gov
Robert Hope Head of Children's Services	555-7811	rhope@mpl.gov
Wayne Lewis Circulation Services Supervisor	555-9001	wlewis@mpl.gov
Bill Forth Interlibrary Loan	555-9391	bforth@mpl.gov

Commenting JavaScript Code

- Commenting your code is an important programming practice



```
<script type="text/javascript">
   function showEM(userName, emServer) {
      /*
         The showEM() function displays a link to the user's e-mail address.
         The text of the user and e-mail server names are entered in
         reverse order to thwart e-mail harvesters.
      */

      userName = stringReverse(userName); // reverse the text of the userName parameter
      emServer = stringReverse(emServer); // reverse the text of the emServer parameter

      var emLink = userName + "@" + emServer; // combine the text of userName and emServer
      document.write("<a href='mailto:" + emLink + "'>");
      document.write(emLink);
      document.write("</a>");
   }
</script>
```

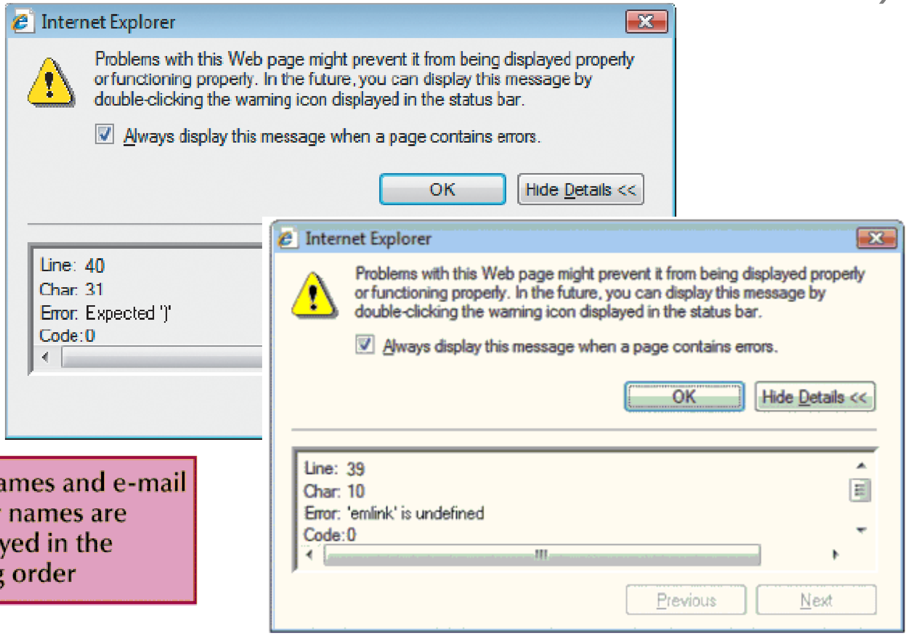
Using Comments to Hide JavaScript Code

```
<script type="text/javascript">
  <!--Hide from nonJavaScript browsers
            JavaScript commands
  // Stop hiding from older browsers -->
  </script>
```

Now all browsers can process JavaScript, so no need to worry about this…

Debugging Your JavaScript Programs

- Debugging is the process of searching code to locate a source of trouble
- There are three types of errors:
  - Load-time errors (e.g., syntax error, similar to compiler error)
  - Run-time errors
  - Logical errors



Internet Explorer

Problems with this Web page might prevent it from being displayed properly or functioning properly. In the future, you can display this message by double-clicking the warning icon displayed in the status bar.

☑ Always display this message when a page contains errors.

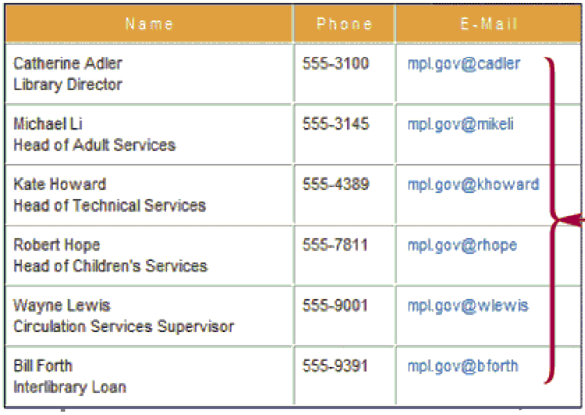
```
Line: 40
Char: 31
Error: Expected ')'
Code: 0
```

Internet Explorer

Problems with this Web page might prevent it from being displayed properly or functioning properly. In the future, you can display this message by double-clicking the warning icon displayed in the status bar.

☑ Always display this message when a page contains errors.

```
Line: 39
Char: 10
Error: 'emlink' is undefined
Code: 0
```



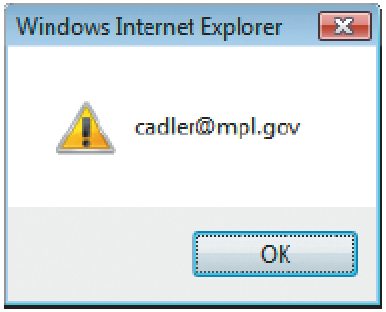
Name	Phone	E-Mail
Catherine Adler Library Director	555-3100	mpl.gov@cadler
Michael Li Head of Adult Services	555-3145	mpl.gov@mikeli
Kate Howard Head of Technical Services	555-4389	mpl.gov@khoward
Robert Hope Head of Children's Services	555-7811	mpl.gov@rhope
Wayne Lewis Circulation Services Supervisor	555-9001	mpl.gov@wlewis
Bill Forth Interlibrary Loan	555-9391	mpl.gov@bforth

usernames and e-mail server names are displayed in the wrong order

Debugging Your JavaScript Programs

- Modular code entails breaking up a program’s different tasks into smaller, more manageable chunks
- An alert dialog box is a dialog box generated by JavaScript that displays a text message with an OK button

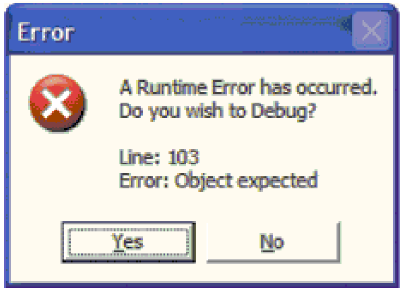
```
alert(text);
```



cadler@mpl.gov

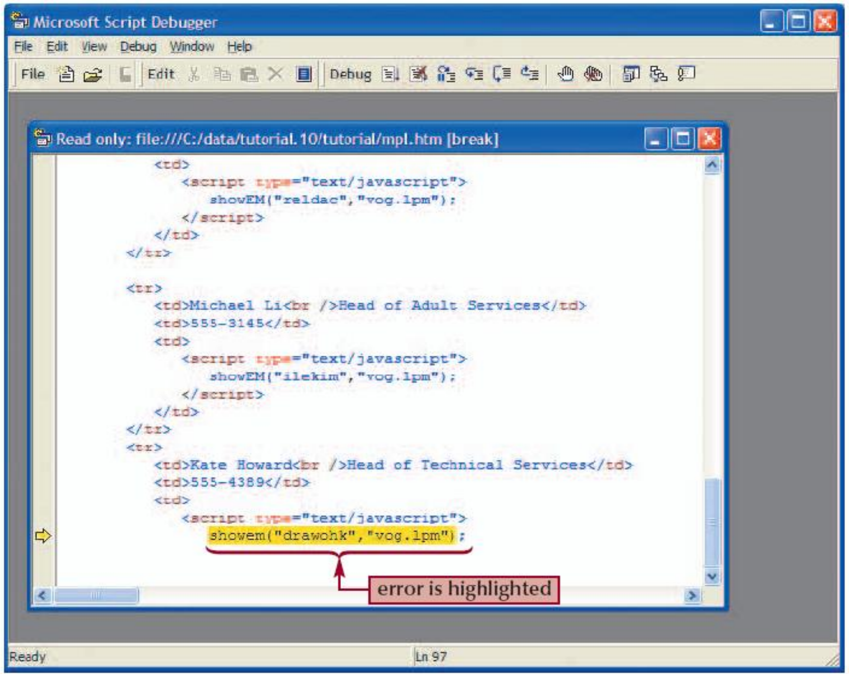
Debugging Your JavaScript Programs

- Microsoft offers the Microsoft Script Debugger
- Firefox also provides the Firefox Error Console



A Runtime Error has occurred.
Do you wish to Debug?

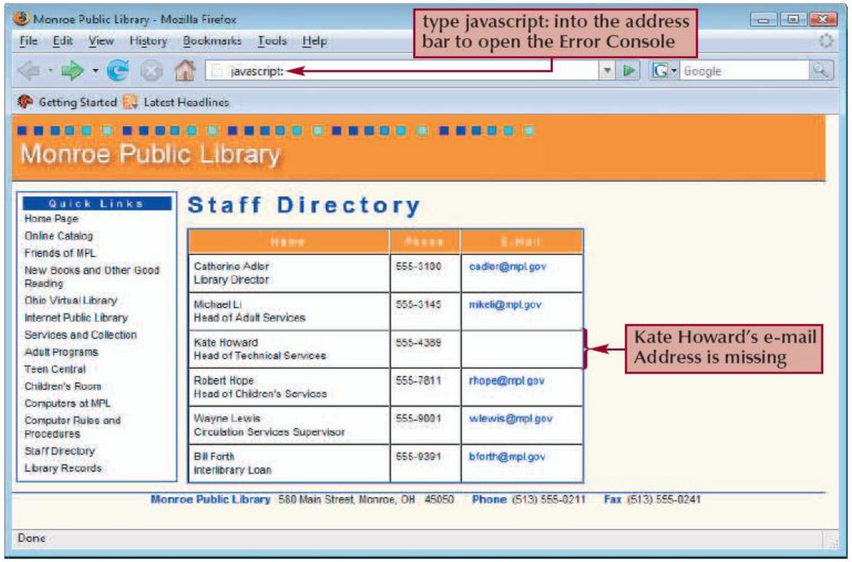
Line: 103
Error: Object expected



Microsoft Script Debugger
File Edit View Debug Window Help
File Edit Debug
Read only: file:///C:/data/tutorial.10/tutorial/mpl.htm [break]

```
      <td>
         <script type="text/javascript">
            showEM("reldac","vog.lpm");
         </script>
      </td>
   </tr>

   <tr>
      <td>Michael Li<br />Head of Adult Services</td>
      <td>555-3145</td>
      <td>
         <script type="text/javascript">
            showEM("ilekim","vog.lpm");
         </script>
      </td>
   </tr>
   <tr>
      <td>Kate Howard<br />Head of Technical Services</td>
      <td>555-4389</td>
      <td>
         <script type="text/javascript">
            showem("drawohk","vog.lpm");
```

error is highlighted



Monroe Public Library - Mozilla Firefox
File Edit View History Bookmarks Tools Help
javascript:
Google
Getting Started
Latest Headlines

type javascript: into the address bar to open the Error Console

Monroe Public Library

Quick Links
Home Page
Online Catalog
Friends of MPL
New Books and Other Good Reading
Ohio Virtual Library
Internet Public Library
Services and Collection
Adult Programs
Teen Central
Children's Room
Computers at MPL
Computer Rules and Procedures
Staff Directory
Library Records

Staff Directory

Name	Phone	E-Mail
Catherine Adler Library Director	555-3100	cadler@mpl.gov
Michael Li Head of Adult Services	555-3145	mikeli@mpl.gov
Kate Howard Head of Technical Services	555-4389	
Robert Hope Head of Children's Services	555-7811	rhope@mpl.gov
Wayne Lewis Circulation Services Supervisor	555-9001	wlewis@mpl.gov
Bill Forth Interlibrary Loan	555-9391	bforth@mpl.gov

Kate Howard's e-mail Address is missing

Monroe Public Library 580 Main Street, Monroe, OH 45050 **Phone** (513) 555-0211 **Fax** (513) 555-0241

JavaScript Events

http://www.w3schools.com/js

```
<html><head>
<script type="text/javascript">
  function displayDate()
  {
    document.getElementById("demo").innerHTML=Date();
  }
</script>
</head>
<body>
  <h1>My First Web Page</h1>
  <p id="demo"></p>
  <button type="button" onclick="displayDate()">
      Display Date</button>
</body>
</html>
```

```
onLoad, onUnload, onSubmit, onFocus,
onBlur, onChange, onMouseOver
```

JavaScript Tutorial

http://www.w3schools.com/js

JS Statements
Variables
Operators
Comparisons
If..Else
Switch
Popup Boxes
Loops (for, while, break …)
Events
Try … Catch
Throw
JS Objects (String, date, array, math …)
JS Advanced