JavaScript Intro – Outline

- What is it good for?
- What does it look like?
- Is it Java?
- Example Usage
What's JavaScript good for?

• Client-side computation?

• Server-side computation?

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns = "http://www.w3.org/1999/xhtml">
<head><title>Printing Multiple Lines</title></head>

<script type = "text/javascript">
  <!--
  document.writeln("<dl>Welcome to<br />JavaScript<br />
  "/javascript"");
  // -->
</script>

<body>
</body>
</html>
```
Java vs. JavaScript

Object-oriented?

Add methods/properties to objects at run-time?

Variable typing?

C-like expressions, control?
Addition / Strings Example – Part 1

Addition / Strings Example – Part 2
Objects and Control Flow – Part 1

<html xmlns = "http://www.w3.org/1999/xhtml">
<head> 
<title>Using Relational Operators</title>
<script type = "text/javascript">
<!--
var now = new Date(); // current date and time
hour = now.getHours(); // current hour (0-23)

// determine whether it is morning
if ( hour < 12 )
    document.write( "<h1>Good Morning, " );

// determine whether the time is PM
if ( hour >= 12 )
{
    // convert to a 12 hour clock
    hour = hour - 12;

    // determine whether it is before 6 PM
    if ( hour < 6 )
        document.write( "<h1>Good Afternoon, " );

    // determine whether it is after 6 PM
    if ( hour >= 6 )
        document.write( "<h1>Good Evening, " );
    }
    document.writeln( "the date is: " + now + "</h1>" );
    // -->
</script>
</head>
<body>
    <p>Click Refresh (or Reload) to run this script again.</p>
</body>
</html>
Operators and Precedence

<table>
<thead>
<tr>
<th>Operators</th>
<th>Associativity</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>/</td>
<td>left to right</td>
<td>multiplicative</td>
</tr>
<tr>
<td>%</td>
<td>left to right</td>
<td>additive</td>
</tr>
<tr>
<td>+</td>
<td>left to right</td>
<td>additive</td>
</tr>
<tr>
<td>-</td>
<td>left to right</td>
<td>additive</td>
</tr>
<tr>
<td>&lt;</td>
<td>left to right</td>
<td>relational</td>
</tr>
<tr>
<td>&lt;=</td>
<td>left to right</td>
<td>relational</td>
</tr>
<tr>
<td>&gt;=</td>
<td>left to right</td>
<td>relational</td>
</tr>
<tr>
<td>==</td>
<td>left to right</td>
<td>equality</td>
</tr>
<tr>
<td>=</td>
<td>right to left</td>
<td>assignment</td>
</tr>
</tbody>
</table>

Fig. 7.17  Precedence and associativity of the operators discussed so far.

Exercise #1 -- What's the output?

```javascript
var a, b, c;

a = 1;
b = 2;
c = 3;

d = a + b * c;

window.alert("<h1>Begin</h1>");

if (d < 20)
  window.alert("d is okay: "+d);
else
  window.alert("d is too high!:"+ d);
d = d - 3;

document.writeln("<h1>Done. Final d = ""+d""</h1>);
```

Adapted from © 2004 Prentice Hall, Inc. All rights reserved.
Exercise #2  -- What's the output?

```javascript
var x, y, z;

x = 7;
y = 9;
z = "abc";

window.alert(x+y+z);

window.alert(z+y+x);

if (x)
  window.alert("x true");

x = "seven";

window.alert(x+y+z);
```

Exercise #3

- Write a JavaScript snippet to read in a number from the user and output its absolute value.
Exercise #4

• Write a JavaScript snippet to read in three numbers x, y, z and output them in sorted order.