

Standard algebraic equality operator or relational operator	JavaScript equality or relational operator	Sample JavaScript condition	Meaning of JavaScript condition
<i>Equality operators</i>			
=	==	x == y	x is equal to y
≠	!=	x != y	x is not equal to y
<i>Relational operators</i>			
>	>	x > y	x is greater than y
<	<	x < y	x is less than y
≥	>=	x >= y	x is greater than or equal to y
≤	<=	x <= y	x is less than or equal to y

Fig. 6.13 | Equality and relational operators.

The script in Fig. 6.14 uses four `if` statements to display a time-sensitive greeting on a welcome page. The script obtains the local time from the user's computer and converts it from 24-hour clock format (0–23) to a 12-hour clock format (0–11). Using this value, the script displays an appropriate greeting for the current time of day. The script and sample output are shown in Fig. 6.14. Lines 11–13 declare the variables used in the script. Also note that JavaScript allows you to assign a value to a variable when it's declared.

*Creating and Using a New **Date** Object*

Line 12 sets the variable `now` to a new **Date** object, which contains information about the current local time. In Section 6.2, we introduced the `document` object, which encapsulates data pertaining to the current web page. Here, we use JavaScript's built-in `Date` object to acquire the current local time. We create a new object by using the `new` operator followed by the type of the object, in this case `Date`, and a pair of parentheses. Some objects require that arguments be placed in the parentheses to specify details about the object to be created. In

```

1  <!DOCTYPE html>
2
3  <!-- Fig. 6.14: welcome5.html -->
4  <!-- Using equality and relational operators. -->
5  <html>
6      <head>
7          <meta charset = "utf-8">
8          <title>Using Relational Operators</title>
9          <script type = "text/javascript">
10             <!--
11                 var name; // string entered by the user
12                 var now = new Date(); // current date and time
13                 var hour = now.getHours(); // current hour (0-23)
14
15                 // read the name from the prompt box as a string
16                 name = window.prompt( "Please enter your name" );

```

Fig. 6.14 | Using equality and relational operators. (Part I of 2.)

```

17
18     // determine whether it's morning
19     if ( hour < 12 )
20         document.write( "<h1>Good Morning, " );
21
22     // determine whether the time is PM
23     if ( hour >= 12 )
24     {
25         // convert to a 12-hour clock
26         hour = hour - 12;
27
28         // determine whether it is before 6 PM
29         if ( hour < 6 )
30             document.write( "<h1>Good Afternoon, " );
31
32         // determine whether it is after 6 PM
33         if ( hour >= 6 )
34             document.write( "<h1>Good Evening, " );
35     } // end if
36
37     document.writeln( name +
38         ", welcome to JavaScript programming!</h1>" );
39     // -->
40     </script>
41     </head><body></body>
42 </html>

```

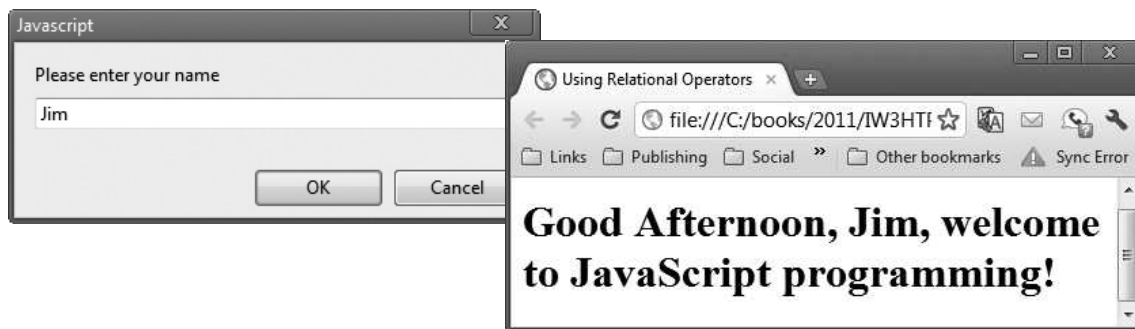


Fig. 6.14 | Using equality and relational operators. (Part 2 of 2.)

this case, we leave the parentheses empty to create a *default* Date object containing information about the current date and time. After line 12 executes, the variable now refers to the new Date object. We did not need to use the new operator when we used the document and window objects because these objects always are created by the browser. Line 13 sets the variable hour to an integer equal to the current hour (in a 24-hour clock format) returned by the Date object's getHours method. Chapter 11 presents a more detailed discussion of the Date object's attributes and methods, and of objects in general. The script uses window.prompt to allow the user to enter a name to display as part of the greeting (line 16).

Decision-Making with the if Statement

To display the correct time-sensitive greeting, the script must determine whether the user is visiting the page during the morning, afternoon or evening. The first if statement (lines