



Exercise 11-4

In addition to specifying the size and position of a window when it first opens, you can also change the size and position of an open window by using methods of the `Window` object. The `resizeTo()` method resizes a window to a specified size, and the `moveTo()` method moves a window to an absolute position. Using these methods with properties of the `Screen` object, you will create a script that resizes and repositions an open window so that it fills the screen.

1. Create a new HTML 5 document in your text editor and use "Maximize Browser Window" as the content of the `<title>` element.
2. Add a form to the document body that includes two command buttons: one that reads "Create New Window" and another that reads "Maximize New Window".
3. Add a script section to the document head.
4. Add the following function for the Create New Window button. This function opens a document named `MaxWindow.html` (which you will create shortly) in a new browser window that is centered in the screen when a user clicks the command button.

```
var maxWindow;  
function createWindow() {  
    var winWidth=300;  
    var winHeight=100;  
    var winLeft = (screen.width-winWidth)/2;  
    var winTop = (screen.height-winHeight)/2;  
    var winOptions = ",width=" + winWidth;  
    winOptions += ",height=" + winHeight;  
    winOptions += ",left=" + winLeft;  
    winOptions += ",top=" + winTop;  
    maxWindow = window.open("MaxWindow.html",  
        "newWindow", winOptions);  
    maxWindow.focus();  
}
```

5. Add the following function for the Maximize New Window button. The first statement in the function uses the `moveTo()` method of the `Window` object to move the window named `maxWindow` (which is created by the `createWindow()`

function) to position 0, 0, which represents the upper-left corner of the screen. The second statement uses the `resizeTo()` method of the `Window` object and the `availWidth` and `availHeight` properties of the `Screen` object to maximize the window. The final statement changes focus to the maximized window.

```
function maximizeWindow() {  
    maxWindow.moveTo(0,0);  
    maxWindow.resizeTo(screen.availWidth,  
        screen.availHeight);  
    maxWindow.focus();  
}
```

6. Save the document as **MaximizeBrowser.html** in the Exercises folder for Chapter 11.
7. Use the W3C Markup Validation Service to validate the **MaximizeBrowser.html** document and fix any errors. Once the document is valid, close it in your text editor.
8. Create a Web page that conforms to the strict DTD, and add the following text and elements to the document body:

```
<p><strong>Resizing and Repositioning  
Example</strong></p>  
<form action="">  
<p><input type="button" value="Close Window"  
    onclick="window.close();" /></p>  
</form>
```
9. Save the document as **MaxWindow.html** in the Exercises folder for Chapter 11.
10. Use the W3C Markup Validation Service to validate the **MaxWindow.html** document and fix any errors. Once the document is valid, close it in your text editor.
11. Open **MaximizeBrowser.html** in your Web browser, and click the **Create New Window** button. The new window should appear centered in the screen. Return to the **MaximizeBrowser.html** file in your Web browser, and click the **Maximize New Window** button. The new window should be resized and repositioned to fill the screen.
12. Close your Web browser windows.



Exercise 11-5

In this exercise, you will create a Web page for a greeting card company. The page will contain links that display images of greeting cards in a separate window. Your Exercises folder for Chapter 11 contains the following greeting card images that you can use for this project: birthday.jpg, halloween.jpg, mothersday.jpg, newyear.jpg, and valentine.jpg.

1. Create a new HTML 5 document in your text editor and use "Gosselin Greeting Cards" as the content of the <title> element.
2. Add the following text and elements to the document body. The onclick events in the links call a function named showCard() that handles the process of displaying each greeting card in a separate window. You create the showCard() function later in this exercise.

```
<h1>Gosselin Greeting Cards</h1>
<h2>All Occasions</h2>
<hr />
<p><a href="valentine.jpg"
onclick="showCard('valentine.jpg');return false">
Valentine's Day</a><br />
<a href="mothersday.jpg"
onclick="showCard('mothersday.jpg');return false">
Mother's Day</a><br />
<a href="halloween.jpg"
onclick="showCard('halloween.jpg');return false">
Halloween</a><br />
<a href="newyear.jpg"
onclick="showCard('newyear.jpg');return false">
New Year</a><br />
<a href="birthday.jpg"
onclick="showCard('birthday.jpg');return false">
Birthday</a></p>
```

3. Add a script section to the document head.
4. Add the following global variable to the script section. This variable will represent the window that displays the greeting card images.

```
var cardWindow;
```


5. Add the following function to the end of the script section. The function opens a new window, centered in the screen, that displays the selected greeting card image.

```
function showCard(linkTarget) {  
    var propertyWidth=400;  
    var propertyHeight=350;  
    var winLeft = (screen.width-propertyWidth)/2;  
    var winTop = (screen.height-propertyHeight)/2;  
    var winOptions =  
        "toolbar=no,menubar=no,location=no, ↵  
        scrollbars=yes,resizable=no";  
    winOptions += ",width=" + propertyWidth;  
    winOptions += ",height=" + propertyHeight;  
    winOptions += ",left=" + winLeft;  
    winOptions += ",top=" + winTop;  
    cardWindow = window.open(linkTarget,  
        "cardInfo", winOptions);  
    cardWindow.focus();  
}
```

6. Save the document as **GreetingCards.html** in the Exercises folder for Chapter 11.
7. Use the W3C Markup Validation Service to validate the **GreetingCards.html** document and fix any errors. Once the document is valid, close it in your text editor, open it in your Web browser, and test the functionality.
8. Close your Web browser window.



Exercise 11-6

You have probably seen Web sites that invite you to add them to your browser's favorites list. With Internet Explorer, you can create a link that automatically adds the Web page to the favorites list by assigning a value of `javascript:window.external.AddFavorite(url, site name)` to the link's href property. Firefox does not contain similar functionality, so you need to use the Navigator object to determine the browser type. In this exercise, you will create a script that contains functionality for adding Course Technology's Web site to a browser's favorites list.

1. Create a new HTML 5 document in your text editor and use "Add to Favorites" as the content of the `<title>` element.
2. Add a script section to the document body.

3. Add the following statements to the script section. The first two statements retrieve the browser's name and version from the Navigator object. The remaining statements create text variables that will be used to create the bookmark link.

```
var browserName = navigator.appName;  
var browserVer = parseInt(navigator.appVersion);  
var linkText = "Add Course Technology  
to your favorites!";  
var url = "http://www.course.com";  
var pageName = "Course Technology";  
var favLink = "";
```

4. Add the following if statement to the end of the script section. The conditional expression determines whether the browser name is equal to "Microsoft Internet Explorer" and whether the browser version is greater than or equal to 4. If so, statements within the if statement build a link that automatically adds the Course Technology Web site to the favorites list in Internet Explorer.

```
if (browserName == "Microsoft Internet Explorer"  
&& browserVer >= 4) {  
    favLink = "<p><a href=\"javascript:window.  
external.AddFavorite(url, pageName)\"<br>  
onmouseover=\"window.status='\"<br>  
linkText + '\"; return true\"<br>  
onmouseout=\"window.status='\"<br>  
\"\"\" + '\"; return true\"<br>  
\">\" + linkText + "</a></p>\"<br>  
document.write(favLink);  
}
```

5. Add the following else clause to the end of the script section to print "Add Course Technology to your favorites! (Ctrl+D)" for all other browsers:

```
else  
    document.write("<p>Add Course Technology  
to your favorites! (Ctrl+D)</p>");
```

6. Save the document as **AddToFavorites.html** in the Exercises folder for Chapter 11, and validate the document with the W3C Markup Validation Service. Once the document is valid, close it in your text editor and then open it in Internet Explorer and test the functionality.
7. Close your Web browser window.