**Common Event Handlers**

JavaScript has a long list of event handlers which it understands. Some of the most commonly used event handlers are:

onclick

onmouseover

onmouseout

onchange

onload

Note: It used to be customary to capitalize interior letters of event handlers,

e.g., onClick and onMouseOver. They still work but will not validate for XHTML.

In addition to these common ones, there are two less common but conceptually important event handlers I feel should also be mentioned:

onfocus

onblur

Again, event handlers would be added to an HTML tag, and would be triggered upon that event occuring to the tag.   
  
In order to support older browsers, most event handlers must be placed within clickable tags, such as A (anchor) and INPUT.

onclick we've already discussed, so I won't repeat myself here.

The onmouseover event occurs when a user passes their mouse over an element.

Example:

<a href="#" onmouseover="alert('Hi');">onmouseover Event</a>

Displayed:

[onmouseover Event](http://fog.ccsf.edu/~srubin/evhand.html)

The onMouseOut event occurs when a user passes their mouse AWAY from an element.

Example:

<a href="#" onmouseout="alert('Bye');">onmouseout Event</a>

Displayed:

[onmouseout Event](http://fog.ccsf.edu/~srubin/evhand.html)

The onmouseover and onMouseOut events are usually used to trigger image switches and custom status bar messages; we'll talk about custom status bar messages a little later in this module.

The onchange event handler is triggered when an element changes its state. The onchange event handler is most commonly used with form elements, such as the SELECT tag pulldown menu.

Example:

<form>

<select name="testPulldown" onchange="alert('You changed me');">

<option value="1">First Choice</option>

<option value="2">Second Choice</option>

<option value="3">Third Choice</option>

</select>

</form>

Displayed:

Top of Form



Bottom of Form

This sort of JavaScript-enhanced interface element is most commonly used for navigational purposes on a web site; we'll talk about that, as well, a little later in this module.

The onload event handler has to do with the loading of HTML pages into a browser window. When a page is completely loaded, when all of its graphics and text and sound files and movies are fully downloaded, the onLoad event is triggered. The onLoad event handler ALWAYS goes in the BODY tag of the HTML page. I am not going to demonstrate this one here; its true purpose is beyond the scope of this class. All I will say is that the onLoad event handler is most commonly used to trigger initialization functions for an HTML page.

The onfocus and onblur event handlers are mostly connected with form elements, and are not commonly used. However, the concepts of FOCUS and BLUR are extremely important, so I thought I should mention them.

When you SELECT something on the computer, you are giving that thing FOCUS. When you DESELECT something, you are giving it BLUR.

Here is a text field:

Top of Form



Bottom of Form

When you click on the text field, you are giving it FOCUS. If there were an onfocus event handler in the text field, it would be triggered when you selected the text field.

When you click AWAY from a selected text field (DESELECTING the element), you are giving that text field BLUR. If there were an onblur event handler in the text field, it would be triggered when you deselected that field.

**Note:** DO NOT try the above onfocus/onblur examples using actual JavaScript code and alert dialog boxes, as this may cause browser crashing.

Focus and blur events occur, not just with form elements, but with browser windows, frames, and, in short, any element which can be selected or deselected. I mention the concept of focus and blur now because we will be dealing with focus methods later in this module, and I want to make certain that you understand what I mean when I say FOCUS.

Again, there are a lot more event handlers. See *"JavaScript: The Definitive Guide"* for more information.