Dynamic HTML:
Event Model

Outline

14.1 Introduction
14.2 Event `onclick`
14.3 Event `onload`
14.4 Error Handling with `onerror`
14.5 Tracking the Mouse with Event `onmousemove`
14.6 Rollovers with `onmouseover` and `onmouseout`
14.7 Form Processing with `onfocus` and `onblur`
14.8 More Form Processing with `onsubmit` and `onreset`
14.9 Event Bubbling
14.10 More DHTML Events
14.11 Web Resources

Objectives

- In this lesson, you will learn:
  - To understand the notion of events, event handlers and event bubbling.
  - To be able to create event handlers that respond to mouse and keyboard events.
  - To be able to use the `event` object to be made aware of and, ultimately, respond to user actions.
  - To understand how to recognize and respond to the most popular events.

14.1 Introduction

- Event model
  - Scripts can respond to user
  - Content becomes more dynamic
  - Interfaces become more intuitive

14.2 Event `onclick`

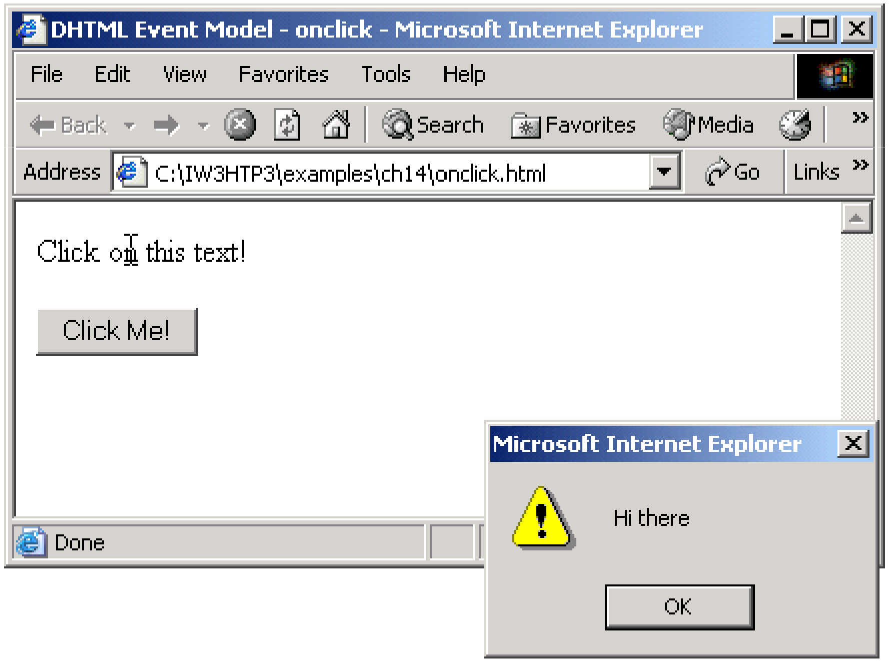
- `onClick`
  - Invoked when user clicks the mouse on a specific item

```
<?xml version = "1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<!-- Fig 14.1: onclick.html          -->
<!-- Demonstrating the onclick event -->

<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>DHTML Event Model - onclick</title>

      <!-- The for attribute declares the script for -->
      <!-- a certain element, and the event for a    -->
      <!-- certain event.                            -->
      <script type = "text/javascript" for = "para"
         event = "onclick">
         <!--
         alert( "Hi there" );
         // -->
      </script>
   </head>

   <body>

```

onclick.html
(1 of 2)

```
      <!-- The id attribute gives a unique identifier -->
      <p id = "para">Click on this text!</p>

      <!-- You can specify event handlers inline -->
      <input type = "button" value = "Click Me!"
         onclick = "alert( 'Hi again' )" />

   </body>
</html>
```

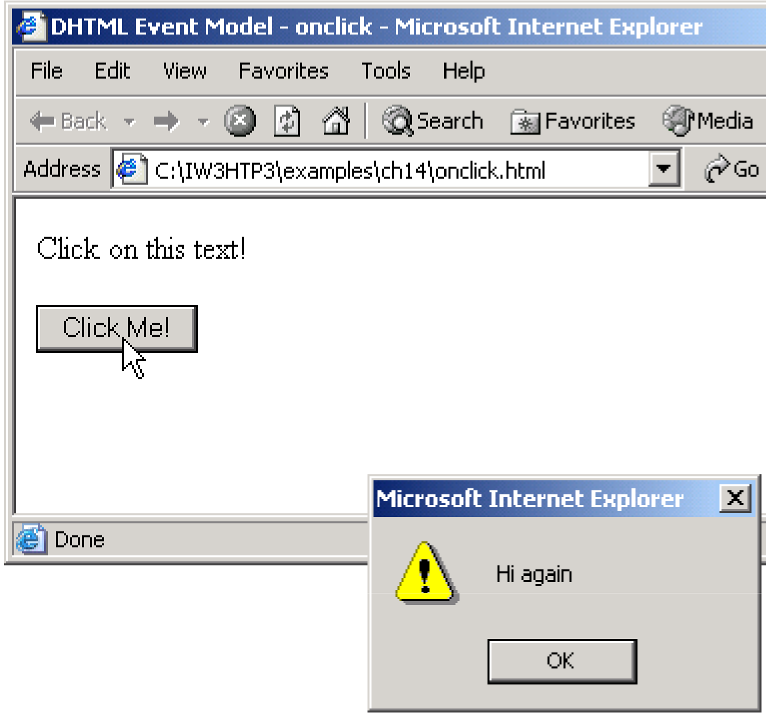
onclick.html
(2 of 2)


DHTML Event Model - onclick - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Back
Search
Favorites
Media
Address
Go
Links
Click Me!
Done
Microsoft Internet Explorer
OK

C:\IW3HTP3\examples\ch14\onclick.html

Click on this text!

Hi there



DHTML Event Model - onclick - Microsoft Internet Explorer

Address C:\IW3HTP3\examples\ch14\onclick.html

Click on this text!

Microsoft Internet Explorer

Hi again

14.3 Event `onload`

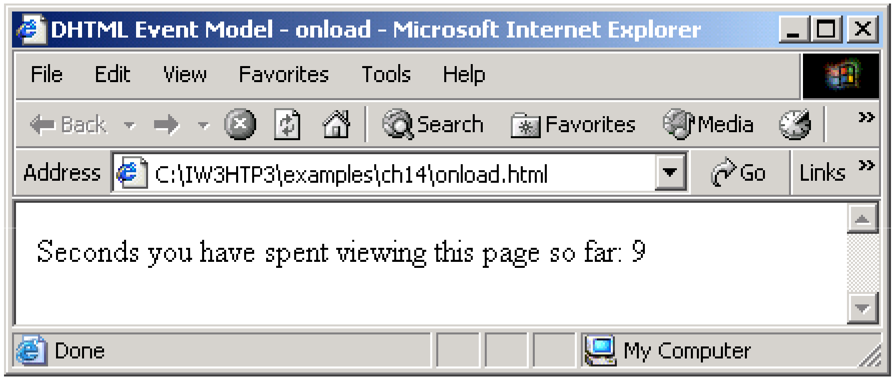
- `onload` event
  - Fires when an element finishes loading
  - Used in the body element
  - Initiates a script after the page loads into the client

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

<!-- Fig. 14.2: onload.html         -->
<!-- Demonstrating the onload event -->

<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>DHTML Event Model - onload</title>
      <script type = "text/javascript">
         <!--
         var seconds = 0;

         function startTimer() {
            // 1000 milliseconds = 1 second
            window.setInterval( "updateTime()", 1000 );
         }

         function updateTime() {
            seconds++;
            soFar.innerText = seconds;
         }
         // -->
      </script>
```

onload.html
(1 of 2)

```
   </head>

   <body onload = "startTimer()">

      <p>Seconds you have spent viewing this page so far:
      <strong id = “soFar”>0</strong></p>

   </body>
</html>
```

onload.html
(2 of 2)


DHTML Event Model - onload - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Back
Search
Favorites
Media
Address
C:\IW3HTP3\examples\ch14\onload.html
Go
Links
Done
My Computer

Seconds you have spent viewing this page so far: 9

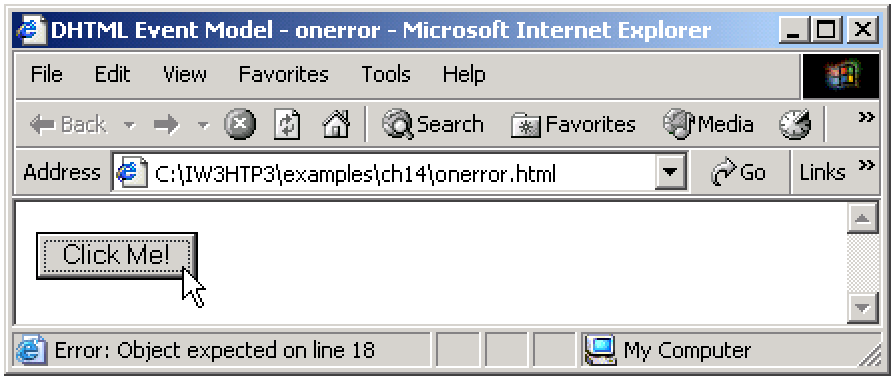
14.4 Error Handling with `onerror`

- `onerror` event
  - Execute specialized error-handling code

onerror.html
(1 of 2)

```
<?xml version = "1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<!-- Fig 14.3: onerror.html            -->
<!-- Demonstrating the onerror event  -->

<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>DHTML Event Model - onerror</title>
      <script type = "text/javascript">
         <!--
         // Specify that if an onerror event is triggered
         // in the window function handleError should execute
         window.onerror = handleError;

         function doThis() {
            alrrt( "hi" ); // alert misspelled, creates an error
         }

         // The ONERROR event passes three values to the
         // function: the name of the error, the url of
         // the file, and the line number.
         function handleError( errType, errURL, errLineNum )
         {
            // Writes to the status bar at the
            // bottom of the window.
            window.status = "Error: " + errType + " on line " +
               errLineNum;

            // Returning a value of true cancels the
            // browser's reaction.
            return true;
         }
         // -->
      </script>
   </head>

   <body>

      <input id = "mybutton" type = "button" value = "Click Me!"
         onclick = "doThis()" />

   </body>
</html>
```

onerror.html
(2 of 2)


DHTML Event Model - onerror - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address C:\IW3HTP3\examples\ch14\onerror.html
Go Links
Click Me!
My Computer

Error: Object expected on line 18

14.5 Tracking the Mouse with Event `onmousemove`

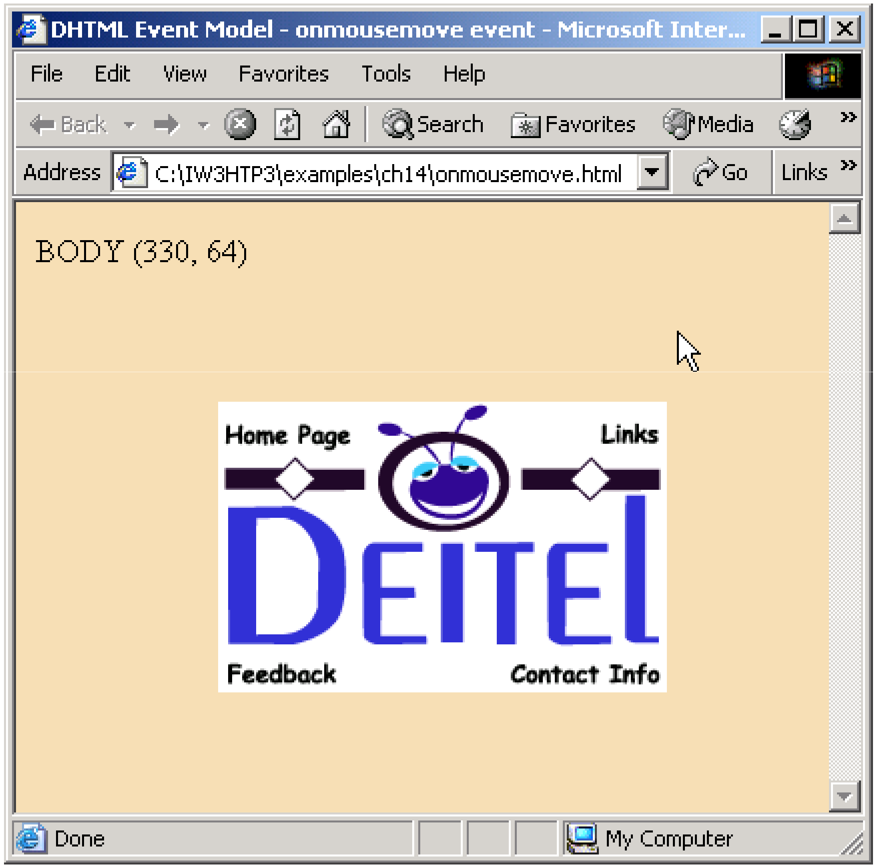
- `onmousemove`
  - Fires repeatedly when the user moves the mouse over the Web page
  - Gives position of the mouse

```
<?xml version = "1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<!-- Fig. 14.4: onmousemove.html          -->
<!-- Demonstrating the onmousemove event  -->

<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>DHTML Event Model - onmousemove event</title>
      <script type = "text/javascript">
         <!--
         function updateMouseCoordinates()
         {
            coordinates.innerText = event.srcElement.tagName +
               " (" + event.offsetX + ", " + event.offsetY + ")";
         }
         // -->
      </script>
   </head>

   <body style = "back-groundcolor: wheat"
      onmousemove = "updateMouseCoordinates()">

```

onmousemove.html
(1 of 2)

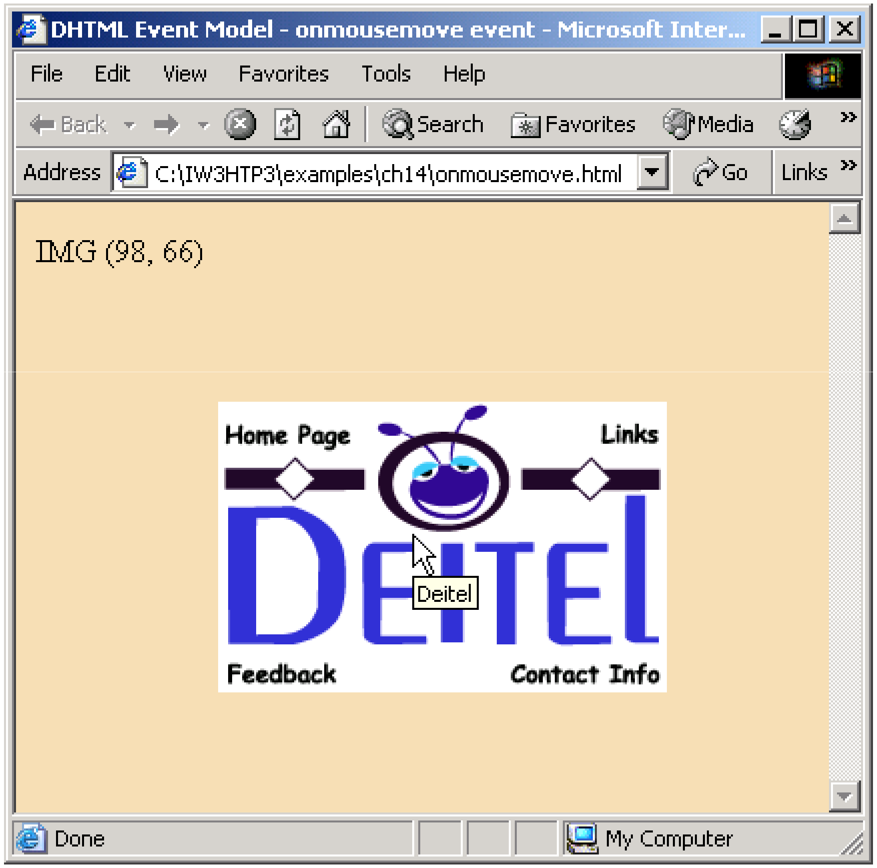
```
      <span id = "coordinates">(0, 0)</span><br />
      <img src = "deitel.gif" style = "position: absolute;
         top: 100; left: 100" alt = "Deitel" />

   </body>
</html>
```

onmousemove.html
(2 of 2)


DHTML Event Model - onmousemove event - Microsoft Inter...
File Edit View Favorites Tools Help
Back Search Favorites Media
Address
Go
Links
Done
My Computer

Address: C:\IW3HTP3\examples\ch14\onmousemove.html

BODY (330, 64)


DHTML Event Model - onmousemove event - Microsoft Inter...
File Edit View Favorites Tools Help
Back Search Favorites Media
Address
Go
Links
Home Page
Links
DEITEL
Feedback
Contact Info
Deitel
Done
My Computer

Address: C:\IW3HTP3\examples\ch14\onmousemove.html

IMG (98, 66)

14.6 Rollovers with `onmouseover` and `onmouseout`

- Two more events fired by mouse movements
  - `onmouseover`
    - Mouse cursor moves over element
  - `Onmouseout`
    - Mouse cursor leaves element

14.6 Rollovers with `onmouseover` and `onmouseout`

Property of `event`	Description
`altkey`	This value is `true` if *Alt* key was pressed when event fired.
`button`	Returns which mouse button was pressed by user (1: left-mouse button, 2: right-mouse button, 3: left and right buttons, 4: middle button, 5: left and middle buttons, 6: right and middle buttons, 7: all three buttons).
`cancelBubble`	Set to `false` to prevent this event from bubbling (see Section 14.9, "Event Bubbling").
`clientX / clientY`	The coordinates of the mouse cursor inside the client area (i.e., the active area where the Web page is displayed, excluding scrollbars, navigation buttons, etc.).
`ctrlKey`	This value is `true` if *Ctrl* key was pressed when event fired.
`offsetX / offsetY`	The coordinates of the mouse cursor relative to the object that fired the event.
`propertyName`	The name of the property that changed in this event.
`recordset`	A reference to a data field's recordset (see Chapter 16, "Data Binding").
`returnValue`	Set to `false` to cancel the default browser action.
`screenX / screenY`	The coordinates of the mouse cursor on the screen coordinate system.
`shiftKey`	This value is `true` if *Shift* key was pressed when event fired.
`srcElement`	A reference to the object that fired the event.
`type`	The name of the event that fired.
`x / y`	The coordinates of the mouse cursor relative to this element's parent element.

Fig. 14.5 Some `event` object properties.

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

<!-- Fig 14.6: onmouseoverout.html     -->
<!-- Events onmouseover and onmouseout -->

<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>
         DHTML Event Model - onmouseover and onmouseout
      </title>
      <script type = "text/javascript">
         <!--
         captionImage1 = new Image();
         captionImage1.src = "caption1.gif";
         captionImage2 = new Image();
         captionImage2.src = "caption2.gif";

         function mOver()
         {
            if ( event.srcElement.id == "tableCaption" ) {
               event.srcElement.src = captionImage2.src;
               return;
            }
```

onmouseoverout.html
(1 of 4)

```

            // If the element which triggered onmouseover has
            // an id, change its color to its id.
            if ( event.srcElement.id )
                event.srcElement.style.color =
                   event.srcElement.id;
         }

         function mOut()
         {
           if ( event.srcElement.id == "tableCaption" ) {
              event.srcElement.src = captionImage1.src;
              return;
           }

           // If it has an id, change the text inside to the
           // text of the id.
           if ( event.srcElement.id )
              event.srcElement.innerText = event.srcElement.id;
         }

         document.onmouseover = mOver;
         document.onmouseout = mOut;
         // -->
      </script>
```

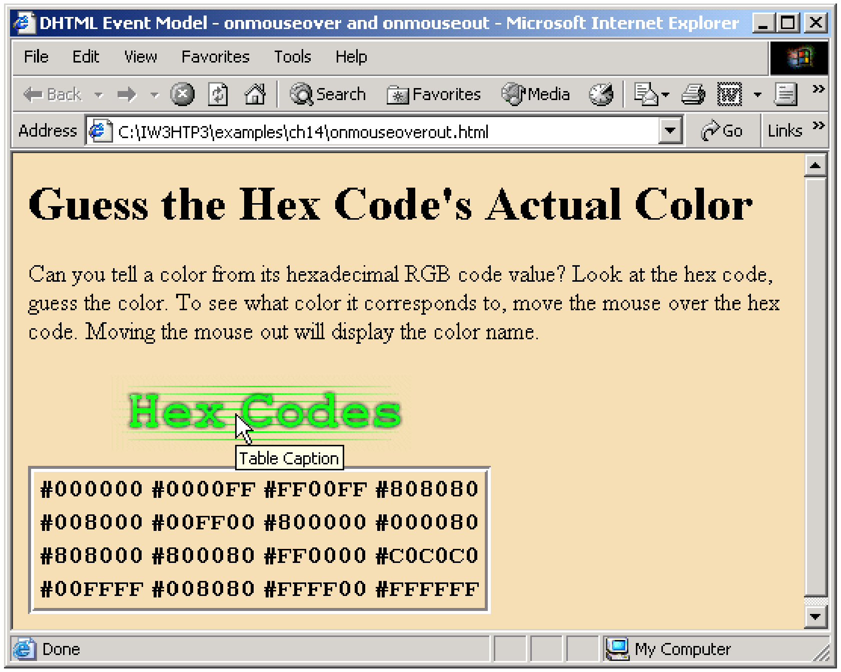
onmouseoverout.html
(2 of 4)

```
</head>

<body style = "background-color: wheat">

   <h1>Guess the Hex Code's Actual Color</h1>

   <p>Can you tell a color from its hexadecimal RGB code
   value? Look at the hex code, guess the color. To see
   what color it corresponds to, move the mouse over the
   hex code. Moving the mouse out will display the color
   name.</p>

   <table style = "width: 50%; border-style: groove;
      text-align: center; font-family: monospace;
      font-weight: bold">

      <caption>
         <img src = "caption1.gif" id = "tableCaption"
            alt = "Table Caption" />
      </caption>

      <tr>
         <td><a id = "Black">#000000</a></td>
         <td><a id = "Blue">#0000FF</a></td>
         <td><a id = "Magenta">#FF00FF</a></td>
```

onmouseoverout.html
(3 of 4)

```html
            <td><a id = "Gray">#808080</a></td>
         </tr>
         <tr>
            <td><a id = "Green">#008000</a></td>
            <td><a id = "Lime">#00FF00</a></td>
            <td><a id = "Maroon">#800000</a></td>
            <td><a id = "Navy">#000080</a></td>
         </tr>
         <tr>
            <td><a id = "Olive">#808000</a></td>
            <td><a id = "Purple">#800080</a></td>
            <td><a id = "Red">#FF0000</a></td>
            <td><a id = "Silver">#C0C0C0</a></td>
         </tr>
         <tr>
            <td><a id = "Cyan">#00FFFF</a></td>
            <td><a id = "Teal">#008080</a></td>
            <td><a id = "Yellow">#FFFF00</a></td>
            <td><a id = "White">#FFFFFF</a></td>
         </tr>
      </table>

   </body>
</html>
```

onmouseoverout.html
(4 of 4)



DHTML Event Model - onmouseover and onmouseout - Microsoft Internet Explorer

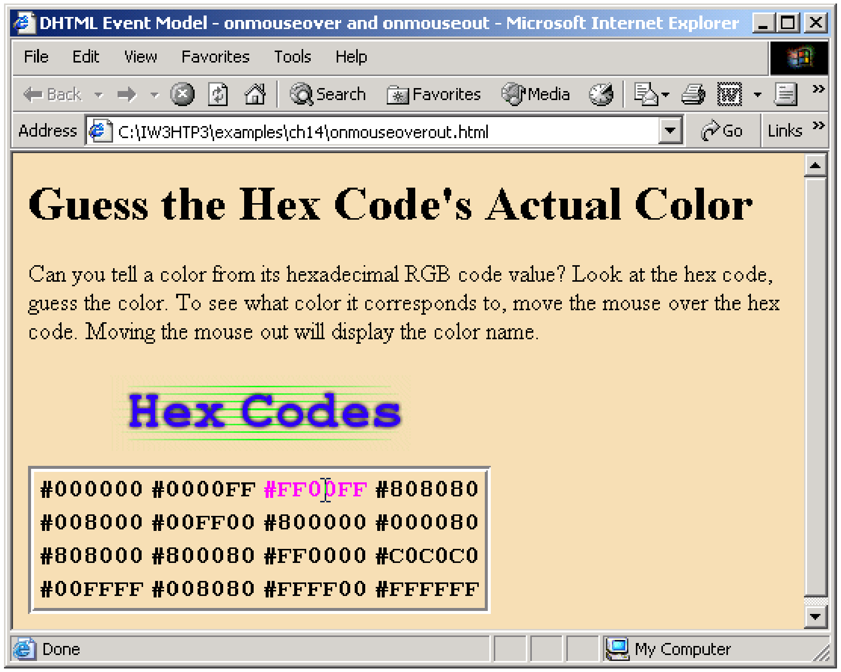
C:\IW3HTP3\examples\ch14\onmouseoverout.html

Guess the Hex Code's Actual Color

Can you tell a color from its hexadecimal RGB code value? Look at the hex code, guess the color. To see what color it corresponds to, move the mouse over the hex code. Moving the mouse out will display the color name.

Hex Codes

#000000	#0000FF	#FF00FF	#808080
#008000	#00FF00	#800000	#000080
#808000	#800080	#FF0000	#C0C0C0
#00FFFF	#008080	#FFFF00	#FFFFFF



DHTML Event Model - onmouseover and onmouseout - Microsoft Internet Explorer

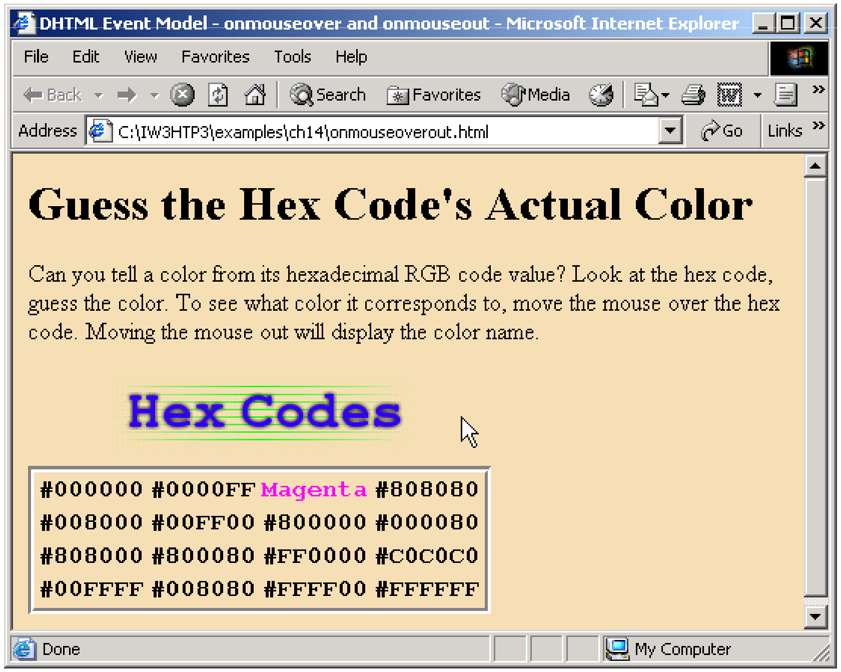
C:\IW3HTP3\examples\ch14\onmouseoverout.html

Guess the Hex Code's Actual Color

Can you tell a color from its hexadecimal RGB code value? Look at the hex code, guess the color. To see what color it corresponds to, move the mouse over the hex code. Moving the mouse out will display the color name.

Hex Codes

#000000	#0000FF	#FF00FF	#808080
#008000	#00FF00	#800000	#000080
#808000	#800080	#FF0000	#C0C0C0
#00FFFF	#008080	#FFFF00	#FFFFFF



DHTML Event Model - onmouseover and onmouseout - Microsoft Internet Explorer

C:\IW3HTP3\examples\ch14\onmouseoverout.html

Guess the Hex Code's Actual Color

Can you tell a color from its hexadecimal RGB code value? Look at the hex code, guess the color. To see what color it corresponds to, move the mouse over the hex code. Moving the mouse out will display the color name.

Hex Codes

#000000	#0000FF	Magenta	#808080
#008000	#00FF00	#800000	#000080
#808000	#800080	#FF0000	#C0C0C0
#00FFFF	#008080	#FFFF00	#FFFFFF

14.7 Form Processing with `onfocus` and `onblur`

- `onfocus` event fires when element gains focus
- `onblur` event fires when element loses focus

```
<?xml version = "1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<!-- Fig. 14.7: onfocusblur.html                   -->
<!-- Demonstrating the onfocus and onblur events   -->

<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>DHTML Event Model - onfocus and onblur</title>
      <script type = "text/javascript">
         <!--
         var helpArray =
            [ "Enter your name in this input box.",
              "Enter your email address in this input box, " +
              "in the format user@domain.",
              "Check this box if you liked our site.",
              "In this box, enter any comments you would " +
              "like us to read.",
              "This button submits the form to the " +
              "server-side script",
              "This button clears the form",
              "This textarea provides context-sensitive " +
              "help. Click on any input field or use the TAB " +
              "key to get more information about the " +
```

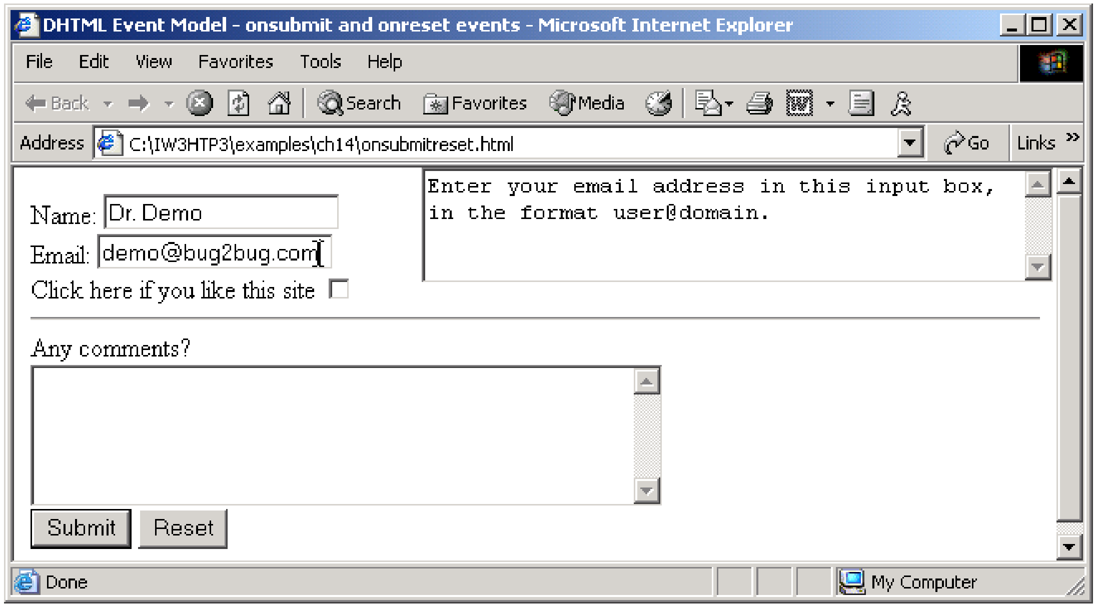
onfocusblur.html
(1 of 3)

```
              "input field." ];

         function helpText( messageNum )
         {
            myForm.helpBox.value = helpArray[ messageNum ];
         }
         // -->
      </script>
   </head>

   <body>

      <form id = "myForm" action = "">
      Name: <input type = "text" name = "name"
         onfocus = "helpText(0)" onblur = "helpText(6)" /><br />
      Email: <input type = "text" name = "email"
         onfocus = "helpText(1)" onblur = "helpText(6)" /><br />
      Click here if you like this site
      <input type = "checkbox" name = "like" onfocus =
         "helpText(2)" onblur = "helpText(6)" /><br /><hr />

      Any comments?<br />
      <textarea name = "comments" rows = "5" cols = "45"
         onfocus = "helpText(3)" onblur = "helpText(6)">
      </textarea><br />
```

onfocusblur.html
(2 of 3)

```
      <input type = "submit" value = "Submit" onfocus =
         "helpText(4)" onblur = "helpText(6)" />
      <input type = "reset" value = "Reset" onfocus =
         "helpText(5)" onblur = "helpText(6)" />

      <textarea name = "helpBox" style = "position: absolute;
         right: 0; top: 0" readonly = “true” rows = "4" cols = "45">
      This textarea provides context-sensitive help.
      Click on any input field or use the Tab key
      to get more information about the input field.</textarea>
      </form>

   </body>
</html>
```

onfocusblur.html
(3 of 3)


DHTML Event Model - onsubmit and onreset events - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address C:\IW3HTP3\examples\ch14\onsubmitreset.html
Go Links
Done
My Computer

Name: Dr. Demo

Email: demo@bug2bug.com

Click here if you like this site □

Any comments?

Submit Reset

```
Enter your email address in this input box,
in the format user@domain.
```

14.8 More Form Processing with `onsubmit` and `onreset`

- `onsubmit` and `onreset` are useful events for processing forms

```
<?xml version = "1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<!-- Fig 14.8: onsubmitreset.html                  -->
<!-- Demonstrating the onsubmit and onreset events -->

<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>
         DHTML Event Model - onsubmit and onreset events
      </title>
      <script type = "text/javascript">
         <!--
         var helpArray =
            [ "Enter your name in this input box.",
              "Enter your email address in this input box, " +
              "in the format user@domain.",
              "Check this box if you liked our site.",
              "In this box, enter any comments you would " +
              "like us to read.",
              "This button submits the form to the " +
              "server-side script",
              "This button clears the form",
              "This textarea provides context-sensitive " +
```

onsubmitreset.html
(1 of 4)

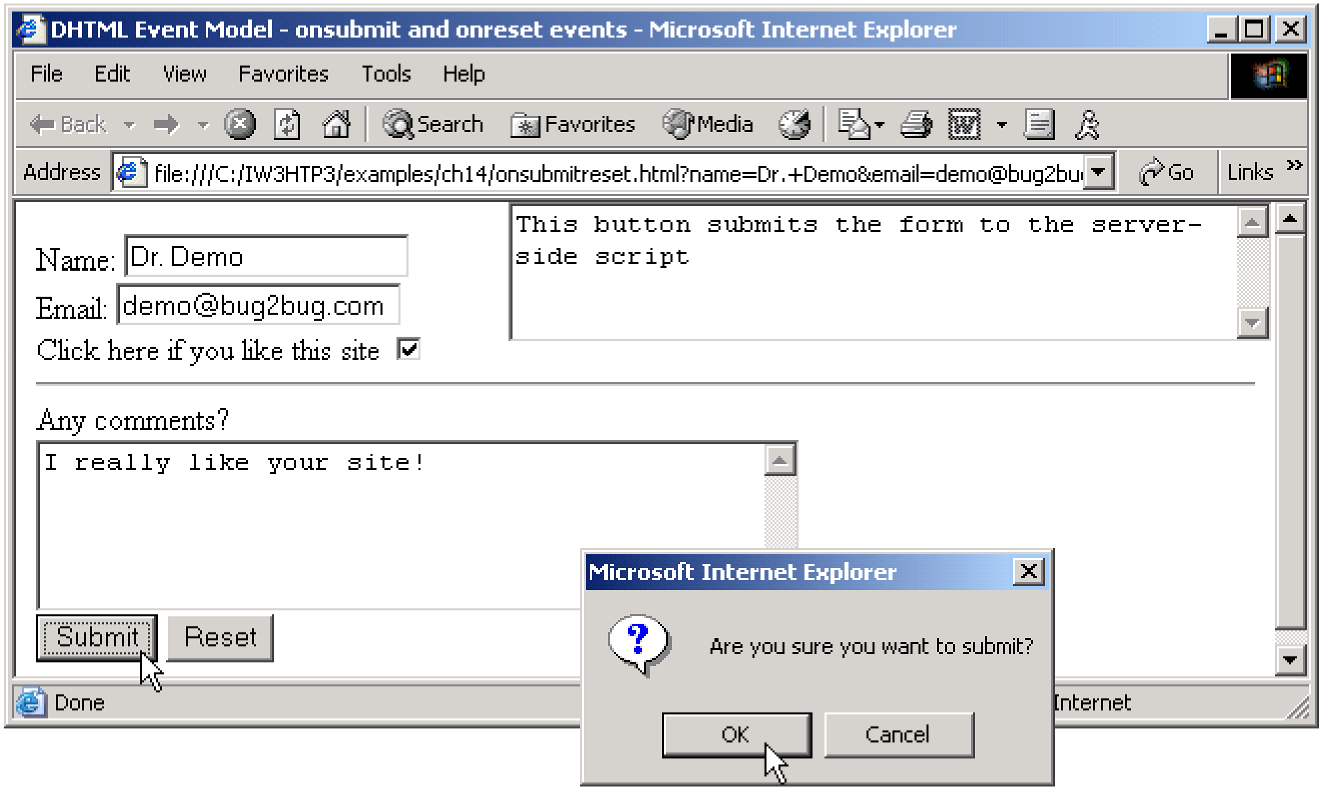
onsubmitreset.html (2 of 4)

```
              "help. Click on any input field or use the Tab " +
              "key to get more information about " +
              "the input field." ];

         function helpText( messageNum )
         {
            myForm.helpBox.value = helpArray[ messageNum ];
         }

         function formSubmit() {
            window.event.returnValue = false;

            if ( confirm ( "Are you sure you want to submit?" ) )
               window.event.returnValue = true;
         }

         function formReset() {
            window.event.returnValue = false;

            if ( confirm( "Are you sure you want to reset?" ) )
               window.event.returnValue = true;
         }
         // -->
      </script>
   </head>

   <body>

      <form id = "myForm" onsubmit = "formSubmit()"
         onreset = "formReset()" action = "">
      Name: <input type = "text" name = "name"
         onfocus =  "helpText(0)" onblur = "helpText(6)" /><br />
      Email: <input type = "text" name = "email"
         onfocus = "helpText(1)" onblur = "helpText(6)" /><br />
      Click here if you like this site
      <input type = "checkbox" name = "like" onfocus =
         "helpText(2)" onblur = "helpText(6)" /><hr />

      Any comments?<br />
      <textarea name = "comments" rows = "5" cols = "45"
         onfocus = "helpText(3)" onblur = "helpText(6)">
      </textarea><br />
      <input type = "submit" value = "Submit" onfocus =
        "helpText(4)"  onblur = "helpText(6)" />
      <input type = "reset" value = "Reset" onfocus =
         "helpText(5)" onblur = "helpText(6)" />

      <textarea name = "helpBox" style = "position: absolute;
         right:0; top: 0" rows = "4" cols = "45">
      This textarea provides context-sensitive help. Click on
```

onsubmitreset.html
(3 of 4)

```
      any input field or use the Tab key to get more
      information about the input field.</textarea>
      </form>

   </body>
</html>
```

onsubmitreset.html
(4 of 4)



DHTML Event Model - onsubmit and onreset events - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address file:///C:/IW3HTP3/examples/ch14/onsubmitreset.html?name=Dr.+Demo&email=demo@bug2bu
Go Links
Done
Internet

Name: Dr. Demo

Email: demo@bug2bug.com

Click here if you like this site ☑

This button submits the form to the server-side script

Any comments?

I really like your site!

Submit Reset

Microsoft Internet Explorer

Are you sure you want to submit?

OK Cancel

14.9 Event Bubbling

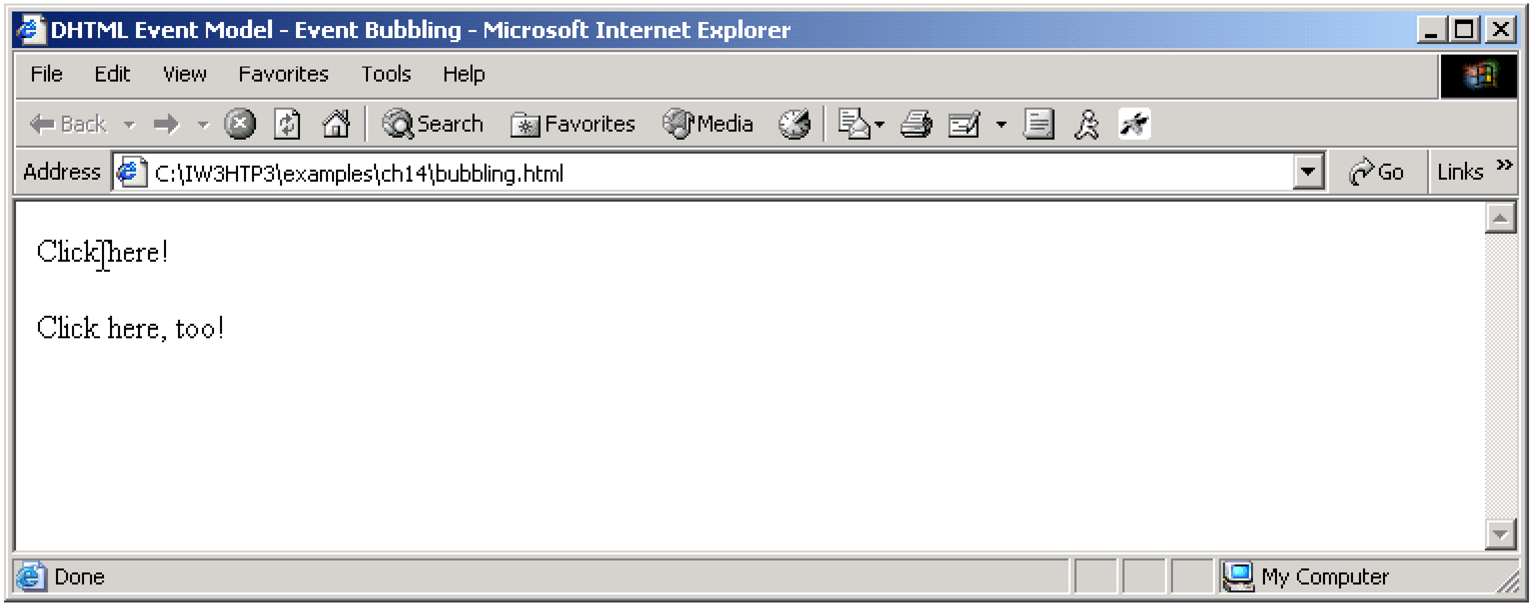
- Crucial part of the event model
- Process whereby events fired in child elements “bubble” up to their parent elements

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

<!-- Fig 14.9: bubbling.html  -->
<!-- Disabling event bubbling -->

<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>DHTML Event Model - Event Bubbling</title>

      <script type = "text/javascript">
         <!--
         function documentClick()
         {
            alert( "You clicked in the document" );
         }

         function paragraphClick( value )
         {
            alert( "You clicked the text" );

            if ( value )
               event.cancelBubble = true;
         }
```

bubbling.html
(1 of 2)

```

         document.onclick = documentClick;
         // -->
      </script>
   </head>

   <body>

      <p onclick = "paragraphClick( false )">Click here!</p>
      <p onclick = "paragraphClick( true )">Click here, too!</p>
   </body>
</html>
```

bubbling.html
(2 of 2)

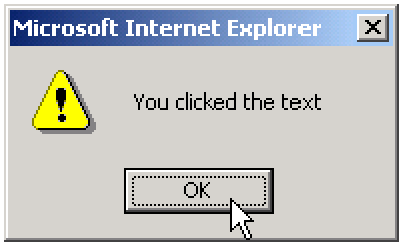


DHTML Event Model - Event Bubbling - Microsoft Internet Explorer

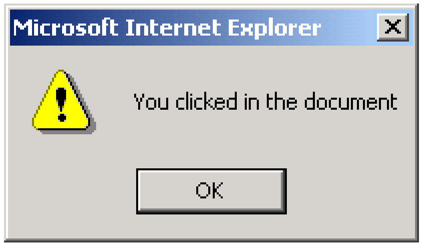
Address C:\IW3HTP3\examples\ch14\bubbling.html

Click here!

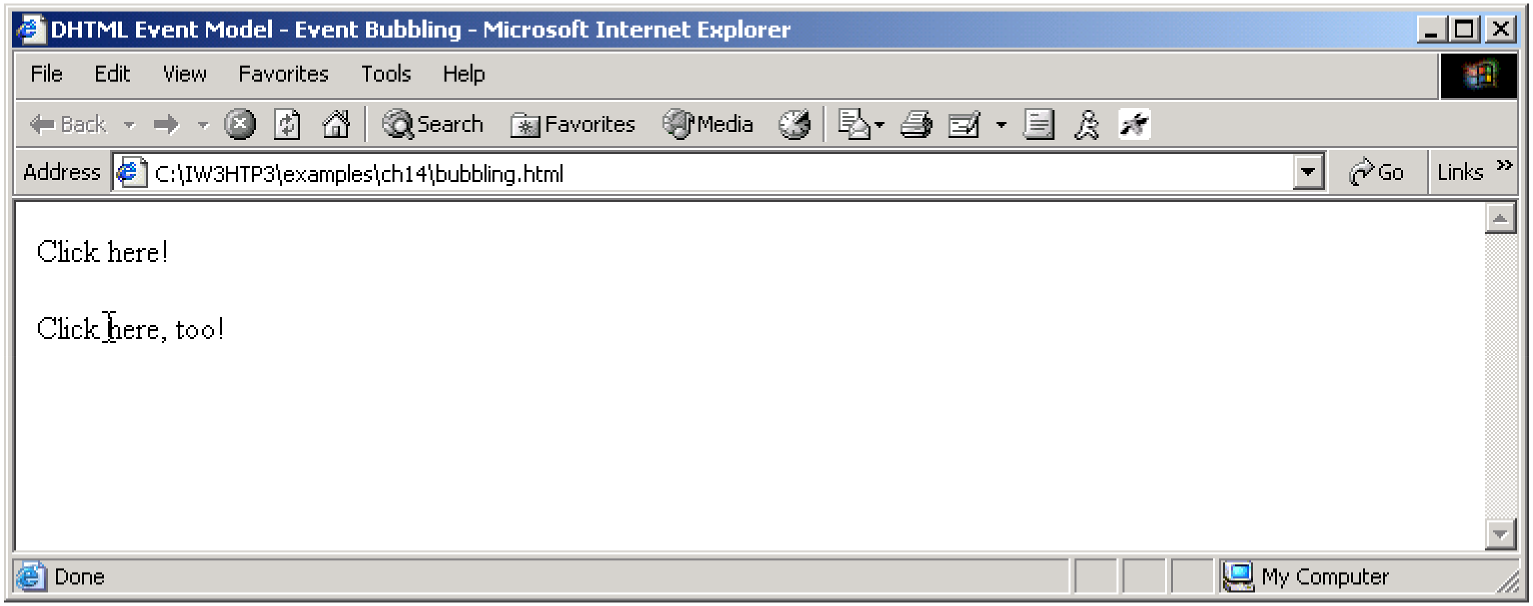
Click here, too!



You clicked the text



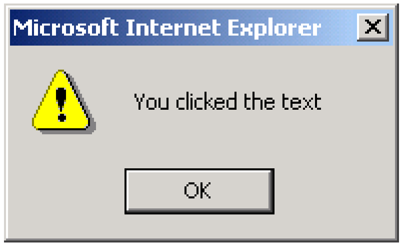
You clicked in the document



Address C:\IW3HTP3\examples\ch14\bubbling.html

Click here!

Click here, too!



You clicked the text

14.10 More DHTML Events

- Remaining DHTML events and their descriptions

14.10 More DHTML Events

Event	Description
Clipboard events	
`onbeforecut`	Fires before a selection is cut to the clipboard.
`onbeforecopy`	Fires before a selection is copied to the clipboard.
`onbeforepaste`	Fires before a selection is pasted from the clipboard.
`oncopy`	Fires when a selection is copied to the clipboard.
`oncut`	Fires when a selection is cut to the clipboard.
`onabort`	Fires if image transfer has been interrupted by user.
`onpaste`	Fires when a selection is pasted from the clipboard.
Data binding events	
`onafterupdate`	Fires immediately after a databound object has been updated.
`onbeforeupdate`	Fires before a data source is updated.
`oncellchange`	Fires when a data source has changed.
`ondataavailable`	Fires when new data from a data source become available.
`ondatasetchanged`	Fires when content at a data source has changed.
`ondatasetcomplete`	Fires when transfer of data from the data source has completed.
`onerrorupdate`	Fires if an error occurs while updating a data field.
`onrowenter`	Fires when a new row of data from the data source is available.
`onrowexit`	Fires when a row of data from the data source has just finished.
`onrowsdelete`	Fires when a row of data from the data source is deleted.
`onrowsinserted`	Fires when a row of data from the data source is inserted.

Fig. 14.10 Dynamic HTML events.

14.10 More DHTML Events

Event	Description
Keyboard events	
`onhelp`	Fires when the user initiates help (i.e., by pressing the *F1* key).
`onkeydown`	Fires when the user pushes down a key.
`onkeypress`	Fires when the user presses a key.
`onkeyup`	Fires when the user ends a key press.
Marquee events	
`onbounce`	Fires when a scrolling `marquee` bounces back in the other direction.
`onfinish`	Fires when a `marquee` finishes its scrolling.
`onstart`	Fires when a `marquee` begins a new loop.
Mouse events	
`oncontextmenu`	Fires when the context menu is shown (right-click).
`ondblclick`	Fires when the mouse is double clicked.
`ondrag`	Fires during a mouse drag.
`ondragend`	Fires when a mouse drag ends.
`ondragenter`	Fires when something is dragged onto an area.
`ondragleave`	Fires when something is dragged out of an area.
`ondragover`	Fires when a drag is held over an area.
`ondragstart`	Fires when a mouse drag begins.
`ondrop`	Fires when a mouse button is released over a valid target during a drag.
`onmousedown`	Fires when a mouse button is pressed down.

Fig. 14.10 Dynamic HTML events.

14.10 More DHTML Events

Event	Description
onmouseup	Fires when a mouse button is released.
Miscellaneous events	
onafterprint	Fires immediately after the document prints.
onbeforeeditfocus	Fires before an element gains focus for editing.
onbeforeprint	Fires before a document is printed.
onbeforeunload	Fires before a document is unloaded (i.e., the window was closed or a link was clicked).
onchange	Fires when a new choice is made in a select element, or when a text input is changed and the element loses focus.
onfilterchange	Fires when a filter changes properties or finishes a transition (see Chapter 15, Dynamic HTML: Filters and Transitions).
onlosecapture	Fires when the releaseCapture method is invoked.
onpropertychange	Fires when the property of an object is changed.
onreadystatechange	Fires when the readyState property of an element changes.
onreset	Fires when a form resets (i.e., the user clicks a reset button).
onresize	Fires when the size of an object changes (i.e., the user resizes a window or frame).
onscroll	Fires when a window or frame is scrolled.
onselect	Fires when a text selection begins (applies to input or textarea).
onselectstart	Fires when the object is selected.
onstop	Fires when the user stops loading the object.
onunload	Fires when a page is about to unload.

Fig. 14.10 Dynamic HTML events.