

Chapter 13 - Dynamic HTML: Object Model and Collections

1

Outline

- 13.1 Introduction
- 13.2 Object Referencing
- 13.3 Collections and children
- 13.4 Dynamic Styles
- 13.5 Dynamic Positioning
- 13.6 Using the frames Collection
- 13.7 navigator Object
- 13.8 Summary of the DHTML Object Model

© 2001 Prentice Hall, Inc. All rights reserved.



2



Outline

Reference.html

```
1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig. 13.1: reference.html -->
6  <!-- Object Model Introduction -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9    <head>
10     <title>Object Model</title>
11
12     <script type = "text/javascript">
13       <!--
14       function start()
15       {
16         alert( pText.innerText );
17         pText.innerText = "Thanks for coming.";
18       }
19       // -->
20     </script>
21
22   </head>
23
24   <body onload = "start()">
25     <p id = "pText">Welcome to our Web page!</p>
26   </body>
27 </html>
```

Function **start** displays an **alert** box containing the value of **pText.innerText**

The **innerText** property of the object refers to the text contained in that element (**Welcome to our Web page!**).

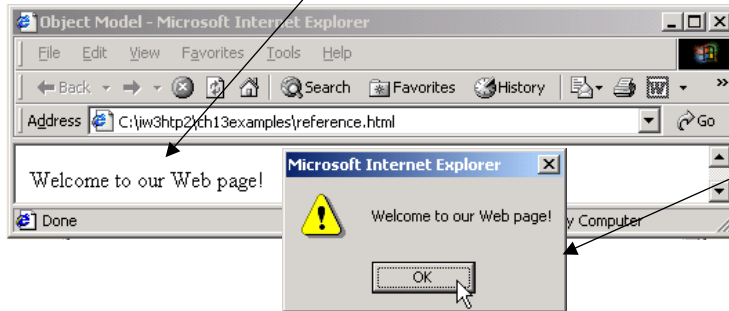
The **innerText** property of the object is dynamically changed (**Thanks for coming.**).

The **onload** event calls the JavaScript **start** function when document loading completes.

© 2001 Prentice Hall, Inc.
All rights reserved.

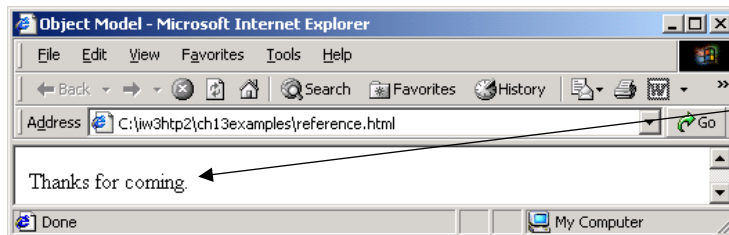


The value of **pText.innerHTML** when the page is first loaded.



An **alert** box displaying the value of **pText.innerHTML**.

Program Output



The value of **pText.innerHTML** after the function **start** is invoked.



All.html

```
1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig 13.2: all.html      -->
6  <!-- Using the all collection -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml"
9    <head>
10      <title>Object Model</title>
11
12      <script type = "text/javascript">
13        <!--
14        var elements = "";
15
16        function start()
17        {
18          for ( var loop = 0; loop < document.all.length; ++loop )
19            elements += "<br />" + document.all[ loop ].tagName;
20
21          pText.innerHTML += elements;
22          alert( elements );
23        }
24        // -->
25      </script>
26    </head>
27
28    <body onload = "start()"
29      <p id = "pText">Elements on this Web page:</p>
30    </body>
31  </html>
```

The **for** loop loops through the elements of the **all** collection and display each element's name.

The **length** property of the **all** collection specifies the number of elements in the collection.

The name of each XHTML element (given in the **tagName** property) in the collection is appended to **elements**.

The **all** collection is a collection of all the XHTML elements in the page in the order they appear.

The **innerHTML** property is similar to the **innerText** property but can also include XHTML formatting.



An **alert** dialog is displayed with all the names of the XHTML elements in the **all** collection.



The **children** collection for a specific element contains that element's child elements.

© 2001 Prentice Hall, Inc.
All rights reserved.



Outline

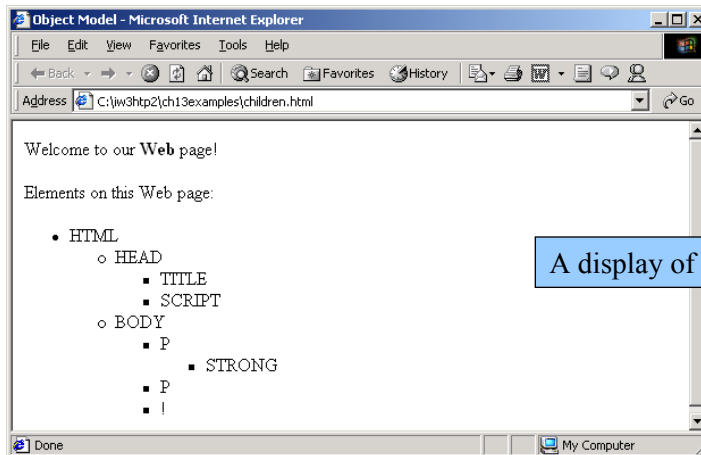
Children.html

```

35     </script>
36 </head>
37
38 <body onload = "child( document.all[ 4 ] );
39     myDisplay.outerHTML += elements;">
40
41     <p>Welcome to our <strong>Web</strong> page!</p>
42
43     <p id = "myDisplay">
44         Elements on this Web page:
45     </p>
46
47 </body>
48 </html>

```

Property **outerHTML** is similar to property **innerHTML** but it includes the enclosing XHTML tags (tags `<p id = "myDisplay">` and `</p>` in this case) as well as the content inside them.



Program Output

A display of all the elements in a document.

© 2001 Prentice Hall, Inc.
All rights reserved.



Outline

Dynamicstyle.html

```

1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3      "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig. 13.4: dynamicstyle.html -->
6  <!-- Dynamic Styles -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9      <head>
10         <title>Object Model</title>
11
12         <script type = "text/javascript">
13             <!--
14             function start()
15             {
16                 var inputColor = prompt(
17                     "Enter a color name for the " +
18                     "background of this page", "" );
19                 document.body.style.backgroundColor = inputColor;
20             }
21             // -->
22         </script>
23     </head>
24
25     <body onload = "start()">
26         <p>Welcome to our Web site!</p>
27     </body>
28 </html>

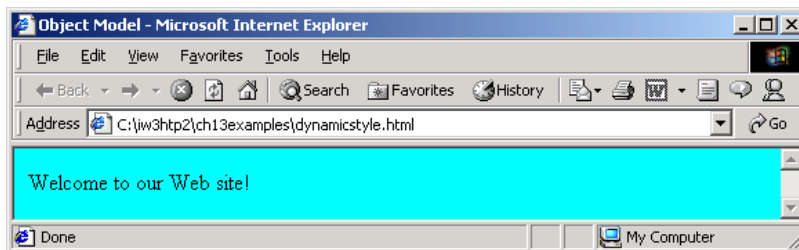
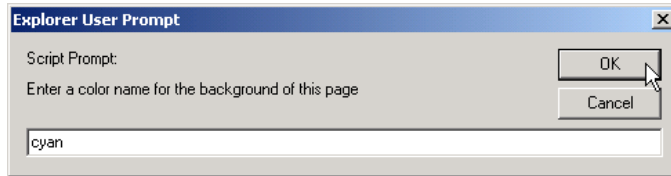
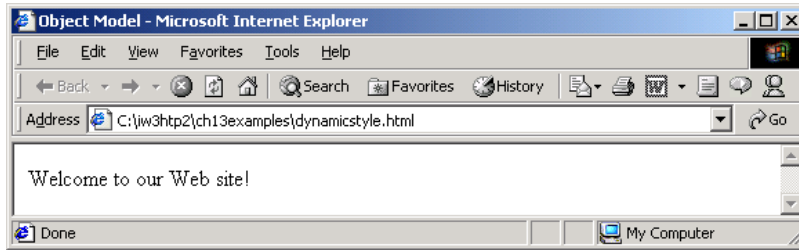
```

Function **start** prompts the user to enter a color and then sets the background to that color.

A **prompt** dialog box is displayed prompting the user to input a color. The color is stored in variable **inputColor**.

The input color is set as the background color.

© 2001 Prentice Hall, Inc.
All rights reserved.



Program Output

Initial Output

Prompt dialog asking user to enter a background color.

Final output with background color selected by the user set.



```
1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig. 13.5: dynamicstyle2.html -->
6  <!-- More Dynamic Styles -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9    <head>
10     <title>Object Model</title>
11
12     <style type = "text/css">
13
14         .bigText    { font-size: 3em;
15                     font-weight: bold }
16
17         .smallText { font-size: .75em }
18
19     </style>
20
21     <script type = "text/javascript">
22         <!--
23         function start()
24         {
25             var inputClass = prompt(
26                 "Enter a className for the text " +
27                 "(bigText or smallText)", "" );
28             pText.className = inputClass;
29         }
30         // -->
31     </script>
32 </head>
33
```

Two style classes for font are defined.

The user is prompted to enter one of the two defined styles to be applied to the text.

Dynamicstyle2.html



Outline

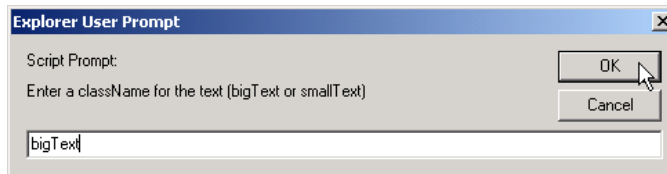
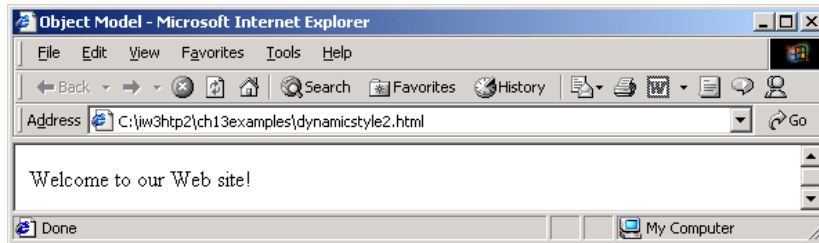
Dynamicstyle2.html

Program Output

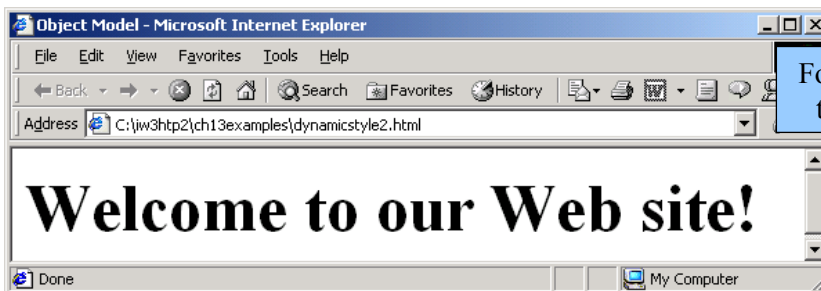
```

34     <body onload = "start()">
35         <p id = "pText">Welcome to our Web site!</p>
36     </body>
37 </html>

```



Dialog prompting user to enter a font style.



Font after user decides to apply the **bigText** style to the text.

© 2001 Prentice Hall, Inc.
All rights reserved.



Outline

Dynamicposition.html

```

1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3      "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig. 13.6: dynamicposition.html -->
6  <!-- Dynamic Positioning -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9      <head>
10         <title>Dynamic Positioning</title>
11
12         <script type = "text/javascript">
13             <!--
14             var speed = 5;
15             var count = 10;
16             var direction = 1;
17             var firstLine = "Text growing";
18             var fontStyle = [ "serif", "sans-serif", "monospace" ];
19             var fontStylecount = 0;
20
21             function start()
22             {
23                 window.setInterval( "run()", 100 );
24             }
25
26             function run()
27             {
28                 count += speed;
29
30                 if ( ( count % 200 ) == 0 ) {
31                     speed *= -1;
32                     direction = !direction;
33

```

Function **start** is used to update the **p** element's content.

Function **setInterval** takes in a function name and how often to run that function.

Function **run** will reverse between increasing and decreasing font size and blue and red font color.

© 2001 Prentice Hall, Inc.
All rights reserved.



Outline

Dynamicposition.
html

```

34      pText.style.color =
35          ( speed < 0 ) ? "red" : "blue" ;
36      firstLine =
37          ( speed < 0 ) ? "Text shrinking" : "Text growing";
38      pText.style.fontFamily =
39          fontStyle[ ++fontStylecount % 3 ];
40  }
41
42      pText.style.fontSize = count / 3;
43      pText.style.left = count;
44      pText.innerHTML = firstLine + "<br /> Font size: " +
45                          count + "px";
46  }
47  // -->
48  </script>
49  </head>
50
51  <body onload = "start()">
52      <p id = "pText" style = "position: absolute; left: 0;
53                              font-family: serif; color: blue">
54          Welcome!</p>
55  </body>
56  </html>

```

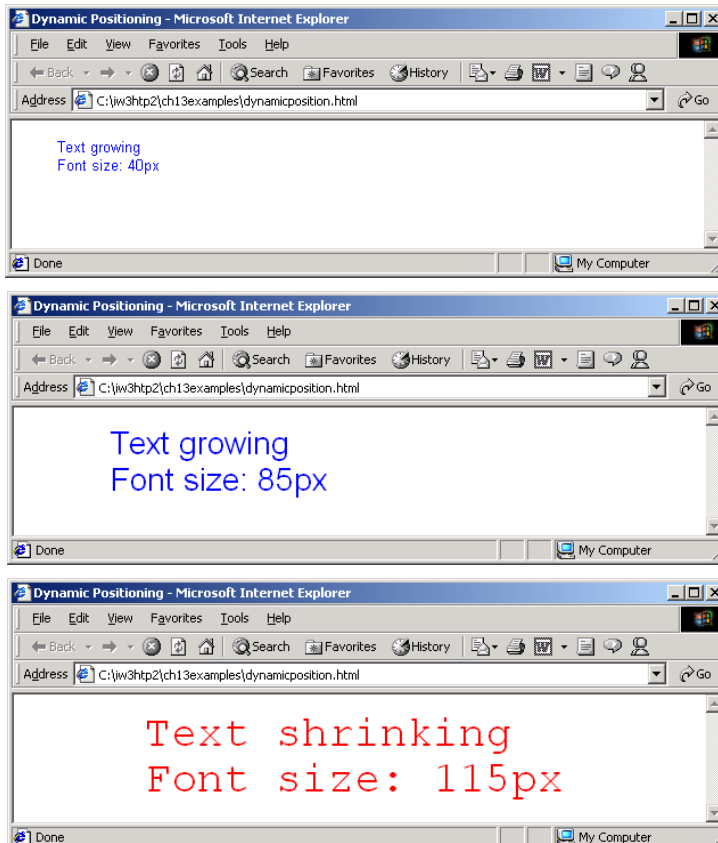
Font size and color will be selected
based on the value of **speed**.

© 2001 Prentice Hall, Inc.
All rights reserved.



Outline

Program Output



Once count reaches a multiple of
200, the size of the text font
starts to shrink by one third.

© 2001 Prentice Hall, Inc.
All rights reserved.



```

1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
3    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
4
5  <!-- Fig. 13.7: index.html      -->
6  <!-- Using the frames collection -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9    <head>
10     <title>Frames collection</title>
11    </head>
12
13    <frameset rows = "100, *">
14      <frame src = "top.html" name = "upper" />
15      <frame src = "" name = "lower" />
16    </frameset>
17
18  </html>

```

The browser window is broken into two horizontal frames.

The top frame (**upper**) displays file **top.html**.

The bottom frame (**lower**) is initially empty.



```

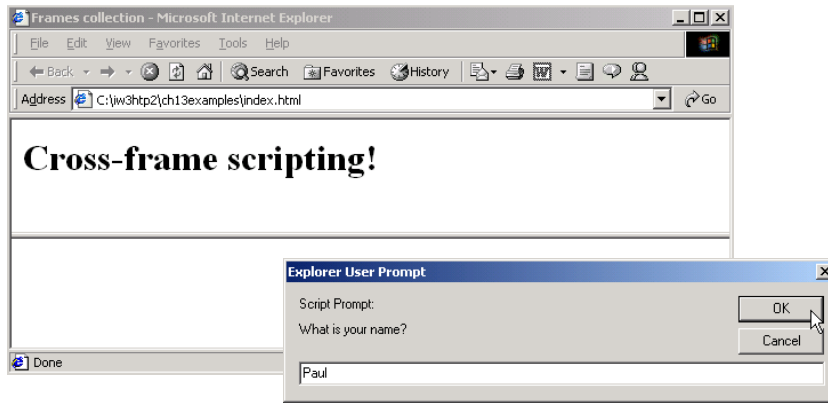
1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig. 13.8: top.html      -->
6  <!-- Cross-frame scripting    -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9    <head>
10     <title>The frames collection</title>
11
12     <script type = "text/javascript">
13       <!--
14       function start()
15       {
16         var text = prompt( "What is your name?", "" );
17         parent.frames( "lower" ).document.write(
18           "<h1>Hello, " + text + "</h1>" );
19       }
20       // -->
21     </script>
22   </head>
23
24   <body onload = "start()">
25     <h1>Cross-frame scripting!</h1>
26   </body>
27 </html>

```

Function **start** takes in a user's name and writes it in a frame in the browser.

The **write** function is used to write text to the frame in the browser.

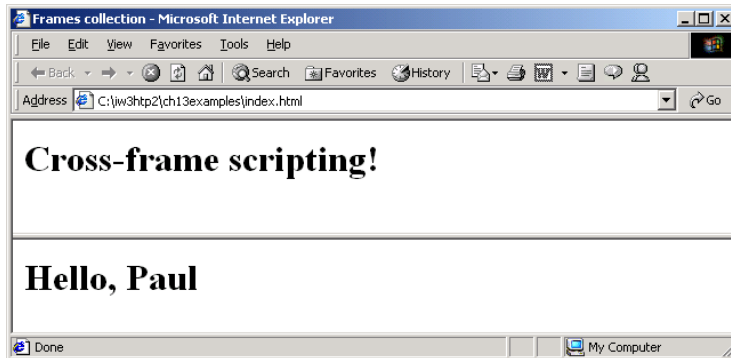
The parent frame of the current frame is first referenced following that the lower frame is referenced.



Program Output

Browser prior to user entering a name.

Dialog prompt for user to enter name.



Browser updated with user name and **Hello** displayed in bottom frame.



```
1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig 13.9: navigator.html -->
6  <!-- Using the navigator object -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9    <head>
10     <title>The navigator Object</title>
11
12     <script type = "text/javascript">
13       <!--
14       function start()
15       {
16         if (navigator.appName=="Microsoft Internet Explorer")
17         {
18           if ( navigator.appVersion.substring( 1, 0 ) >= "4" )
19             document.location = "newIEversion.html";
20           else
21             document.location = "oldIEversion.html";
22         }
23         else
24           document.location = "NSversion.html";
25       }
26       // -->
27     </script>
28   </head>
29
30   <body onload = "start()">
31     <p>Redirecting your browser to the appropriate page,
32     please wait...</p>
33   </body>
34 </html>
```

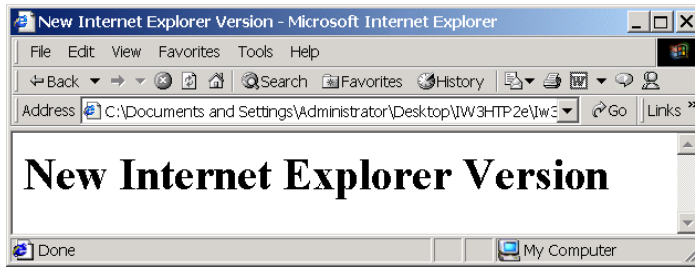
Navigator.html

On load function **start** is called to determine what browser is being used. Based on this a page specific to that browser is displayed.

The browser is Microsoft Internet Explorer.

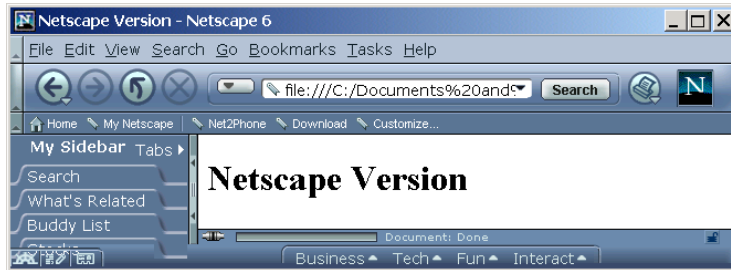
The browser is Netscape.

The function **appVersion** is used to determine which version of IE is used.



Program Output

The document specific for the Microsoft Internet Explorer browser is displayed.



The document specific for the Netscape browser is displayed.

(Netscape Communicator browser window©
1999 Netscape Communications Corporation.
Used with permission. Netscape
Communications has not authorized,
sponsored, endorsed, or approved this
publication and is not responsible for its
content.)

© 2001 Prentice Hall, Inc.
All rights reserved.

13.8 Summary of the DHTML Object Model

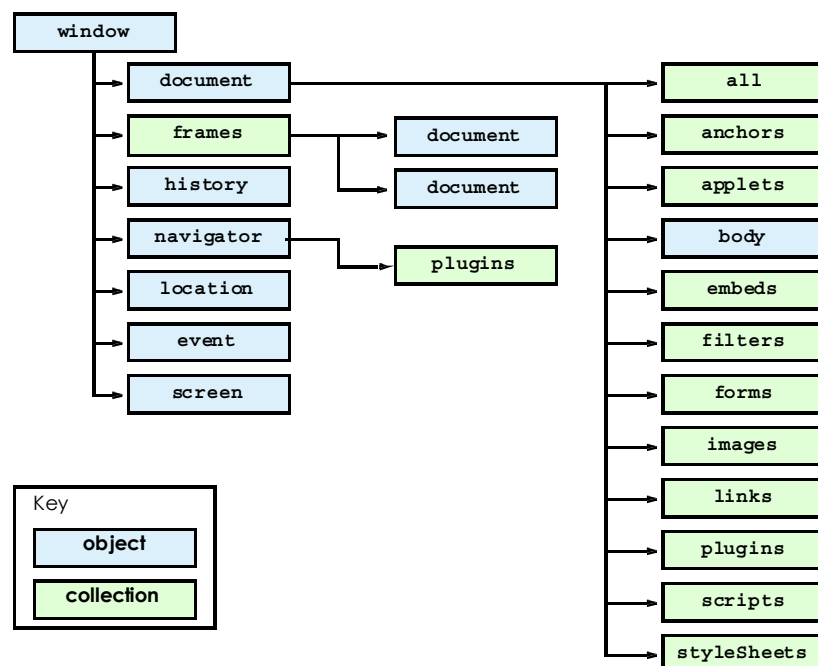


Fig. 13.10 DHTML Object Model.



13.8 Summary of the DHTML Object Model

Object or collection	Description
<i>Objects</i>	
window	This object represents the browser window and provides access to the document object contained in the window . If the window contains frames, a separate window object is created automatically for each frame, to provide access to the document rendered in that frame. Frames are considered to be subwindows in the browser.
document	This object represents the XHTML document rendered in a window . The document object provides access to every element in the XHTML document and allows dynamic modification of the XHTML document.
body	This object provides access to the body element of an XHTML document.
history	This object keeps track of the sites visited by the browser user. The object provides a script programmer with the ability to move forward and backward through the visited sites, but for security reasons does not allow the actual site URLs to be manipulated.
navigator	This object contains information about the Web browser, such as the name of the browser, the version of the browser, the operating system on which the browser is running and other information that can help a script writer customize the user's browsing experience.
location	This object contains the URL of the rendered document. When this object is set to a new URL, the browser immediately switches (navigates) to the new location.
event	This object can be used in an event handler to obtain information about the event that occurred (e.g., the mouse coordinates during a mouse event).
screen	The object contains information about the computer screen for the computer on which the browser is running. Information such as the width and height of the screen in pixels can be used to determine the size at which elements should be rendered in a Web page.
Fig. 13.11 Objects in the Internet Explorer 5.5 object model.	



13.8 Summary of the DHTML Object Model

Object or collection	Description
<i>Collections</i>	
all	Many objects have an all collection that provides access to every element contained in the object. For example, the body object's all collection provides access to every element in the body element of an XHTML document.
anchors	This collection contains all anchor elements (a) that have a name or id attribute. The elements appear in the collection in the order they were defined in the XHTML document.
applets	This collection contains all the applet elements in the XHTML document. Currently, the most common applet elements are Java applets.
embeds	This collection contains all the embed elements in the XHTML document.
forms	This collection contains all the form elements in the XHTML document. The elements appear in the collection in the order they were defined in the XHTML document.
frames	This collection contains window objects that represent each frame in the browser window. Each frame is treated as its own subwindow.
images	This collection contains all the img elements in the XHTML document. The elements appear in the collection in the order they were defined in the XHTML document.
links	This collection contains all the anchor elements (a) with an href property. This collection also contains all the area elements that represent links in an image map.
plugins	Like the embeds collection, this collection contains all the embed elements in the XHTML document.
scripts	This collection contains all the script elements in the XHTML document.
styleSheets	This collection contains styleSheet objects that represent each style element in the XHTML document and each style sheet included in the XHTML document via link .
Fig. 13.11 Objects in the Internet Explorer 5.5 object model.	

