Dynamic HTML: Object Model and Collections

Outline

Objectives

- In this lesson, you will learn:
  - To use the Dynamic HTML Object Model and scripting to create dynamic Web pages.
  - To understand the Dynamic HTML object hierarchy.
  - To use the `all` and `children` collections to enumerate all of the XHTML elements of a Web page.
  - To use dynamic styles and dynamic positioning.
  - To use the `frames` collection to access objects in a separate frame on your Web page.
  - To use the `navigator` object to determine which browser is being used to access your page.

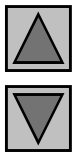
13.1 Introduction

- Dynamic HTML Object Model
  - Allows Web authors to control the presentation of their pages
  - Gives them access to all the elements on their pages
- Web page
  - Elements, forms, frames, tables
  - Represented in an object hierarchy
- Scripting
  - Retrieve and modify properties and attributes

13.2 Object Referencing

- The simplest way to reference an element is by using the element's `id` attribute.
- The element is represented as an object
  - XHTML attributes become properties that can be manipulated by scripting

```
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>
```



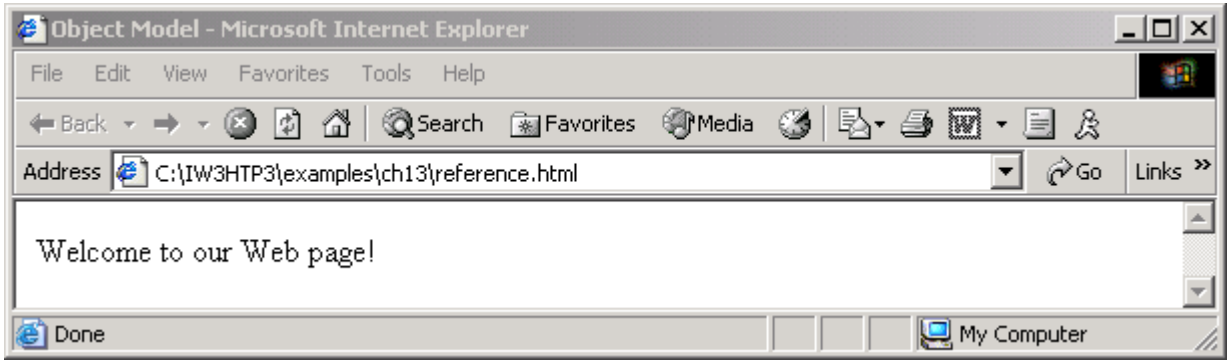
Outline

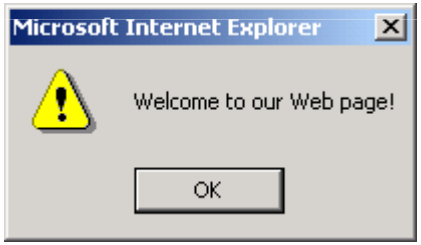
reference.html
(1 of 2)

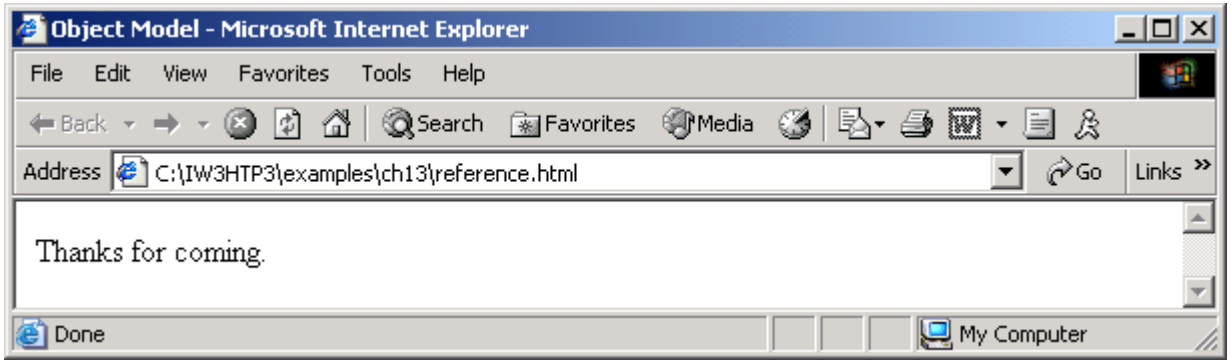
```
23
24      <body onload = "start()">
25         <p id = "pText">Welcome to our Web page!</p>
26      </body>
27   </html>
```


Outline

reference.html
(2 of 2)



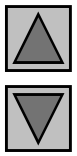




13.3 Collections `all` and `children`

- Collections
  - Arrays of related objects on a page
  - `all`
    - all the XHTML elements in a document
  - `children`
    - Specific element contains that element's child elements

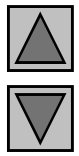
```
1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig 13.2: all.html       -->
6  <!-- Using the all collection -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9     <head>
10       <title>Object 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          // -->
```



Outline

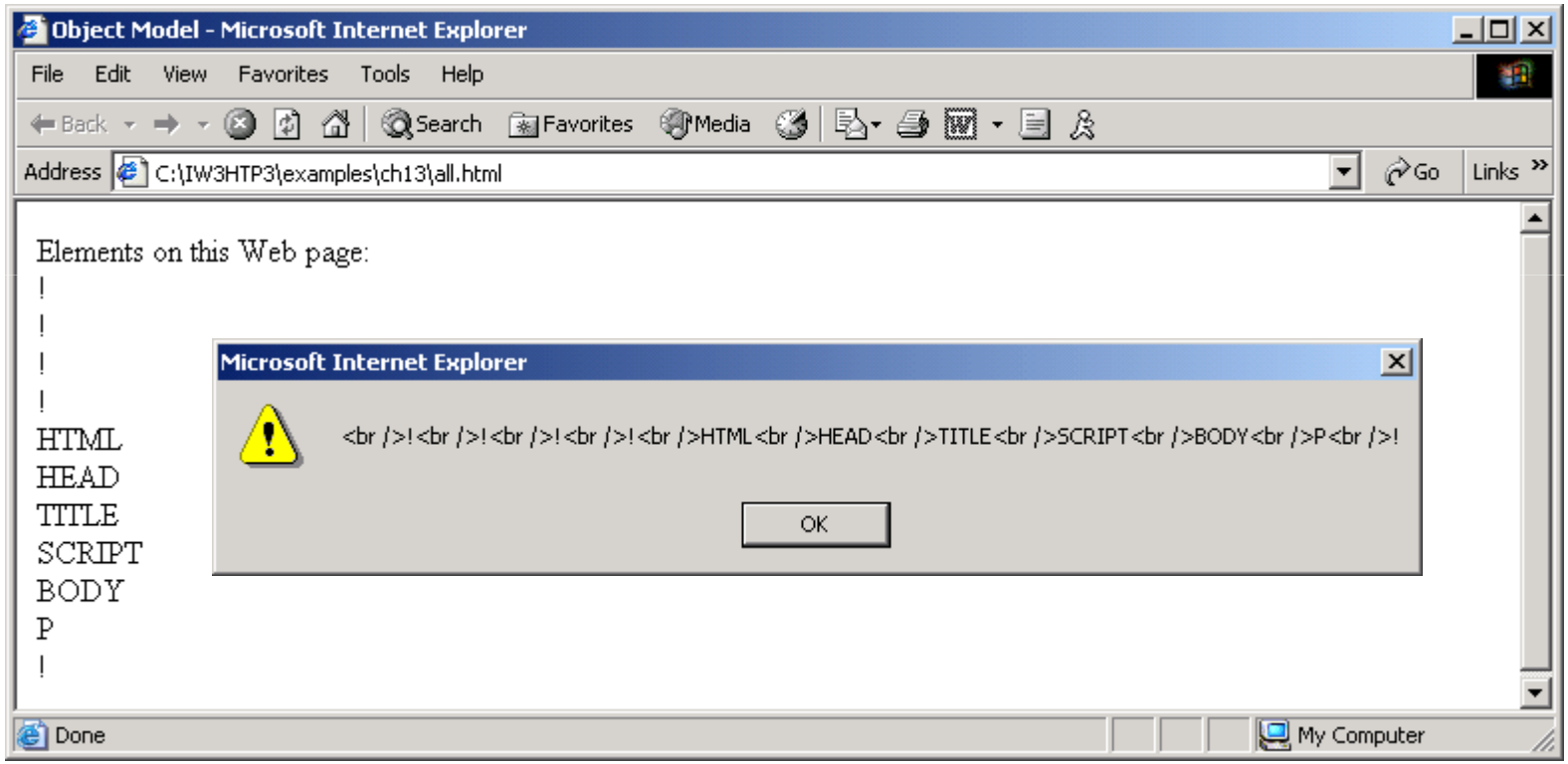
all.html
(1 of 2)

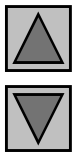
```
25       </script>
26    </head>
27
28    <body onload = "start()">
29       <p id = "pText">Elements on this Web page:</p>
30    </body>
31 </html>
```



Outline

all.html
(2 of 2)



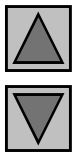


Outline

children.html
(1 of 3)

```
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 13.3: children.html -->
6  <!-- The children 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 = "<ul>";
15
16          function child( object )
17          {
18             var loop = 0;
19
20             elements += "<li>" + object.tagName + "<ul>";
21
```

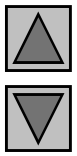
```
22            for ( loop = 0; loop < object.children.length; loop++ )
23            {
24               if ( object.children[ loop ].children.length )
25                  child( object.children[ loop ] );
26               else
27                  elements += "<li>" +
28                     object.children[ loop ].tagName +
29                     "</li>";
30            }
31
32            elements += "</ul>" + "</li>";
33         }
34         // -->
35      </script>
36   </head>
37
```



Outline

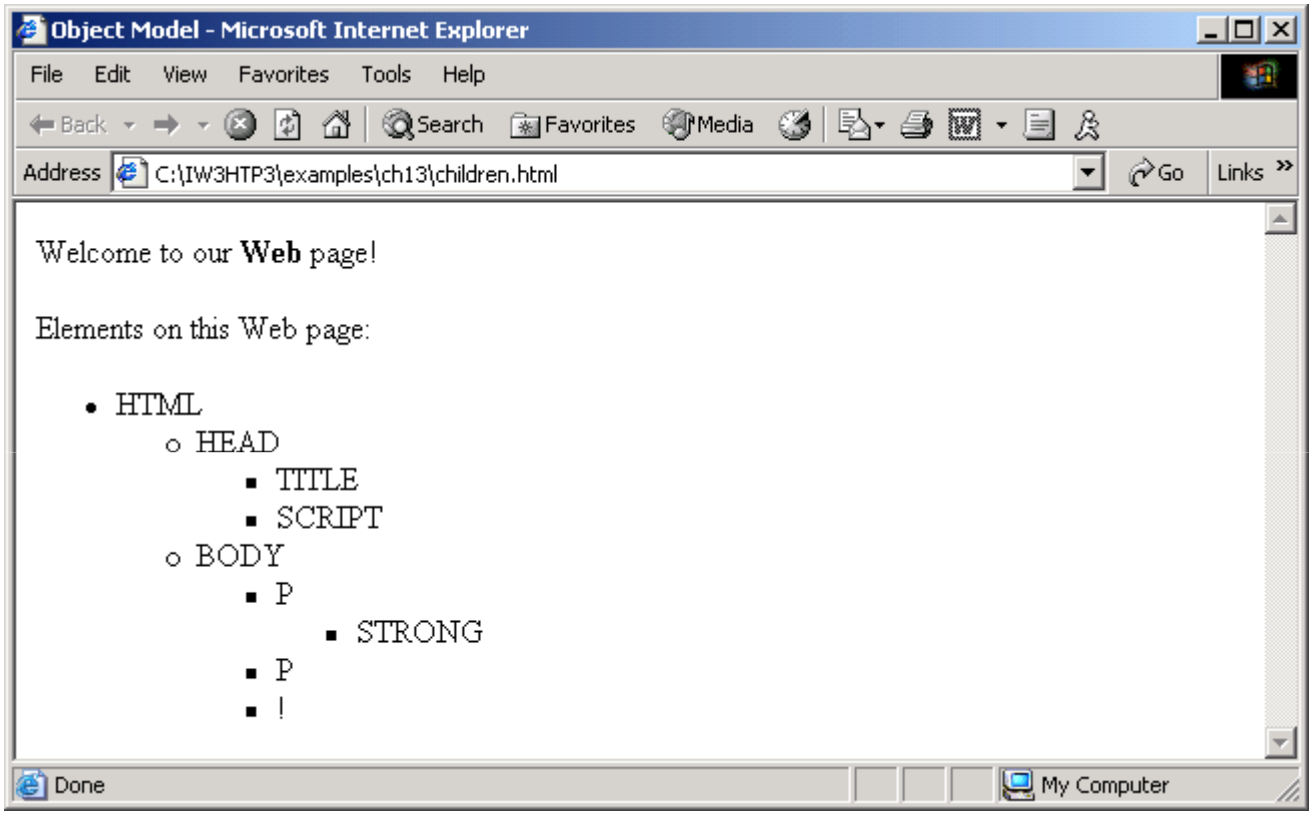
children.html
(2 of 3)

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



Outline

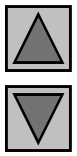
children.html
(3 of 3)


Object Model - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address C:\IW3HTP3\examples\ch13\children.html
Go Links
Welcome to our Web page!
Elements on this Web page:
HTML
HEAD
TITLE
SCRIPT
BODY
P
STRONG
P
!
Done
My Computer

13.4 Dynamic Styles

- Element's style can be changed dynamically
- Dynamic HTML Object Model also allows you to change the `class` attribute

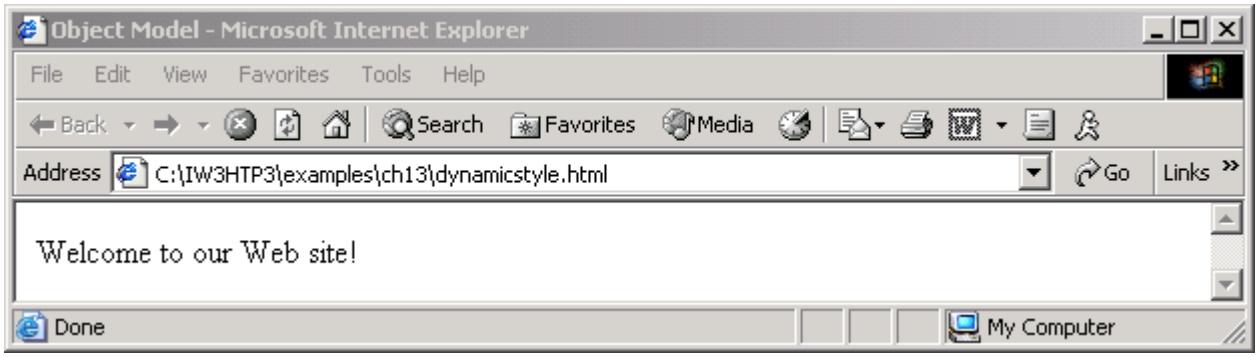
```
1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig. 13.4: dynamicstyle.html -->
6  <!-- Dynamic Styles               -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9     <head>
10       <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>
```

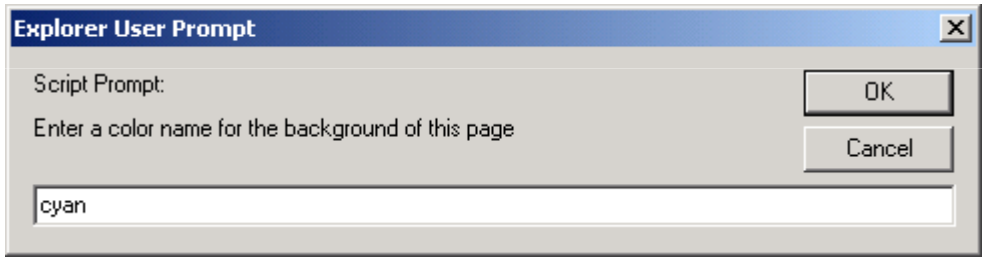


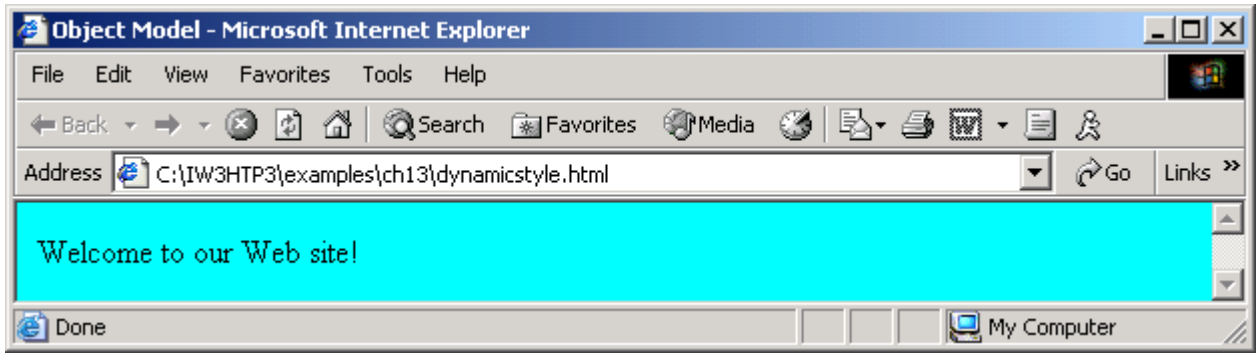
Outline

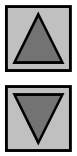
dynamicstyle.html
(1 of 2)

```
24
25     <body onload = "start()">
26        <p>Welcome to our Web site!</p>
27     </body>
28  </html>
```





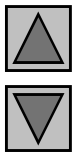




Outline

dynamicstyle.html
(2 of 2)

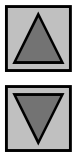
```
1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig. 13.5: dynamicstyle2.html -->
6  <!-- More Dynamic Styles           -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9     <head>
10       <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
```



Outline

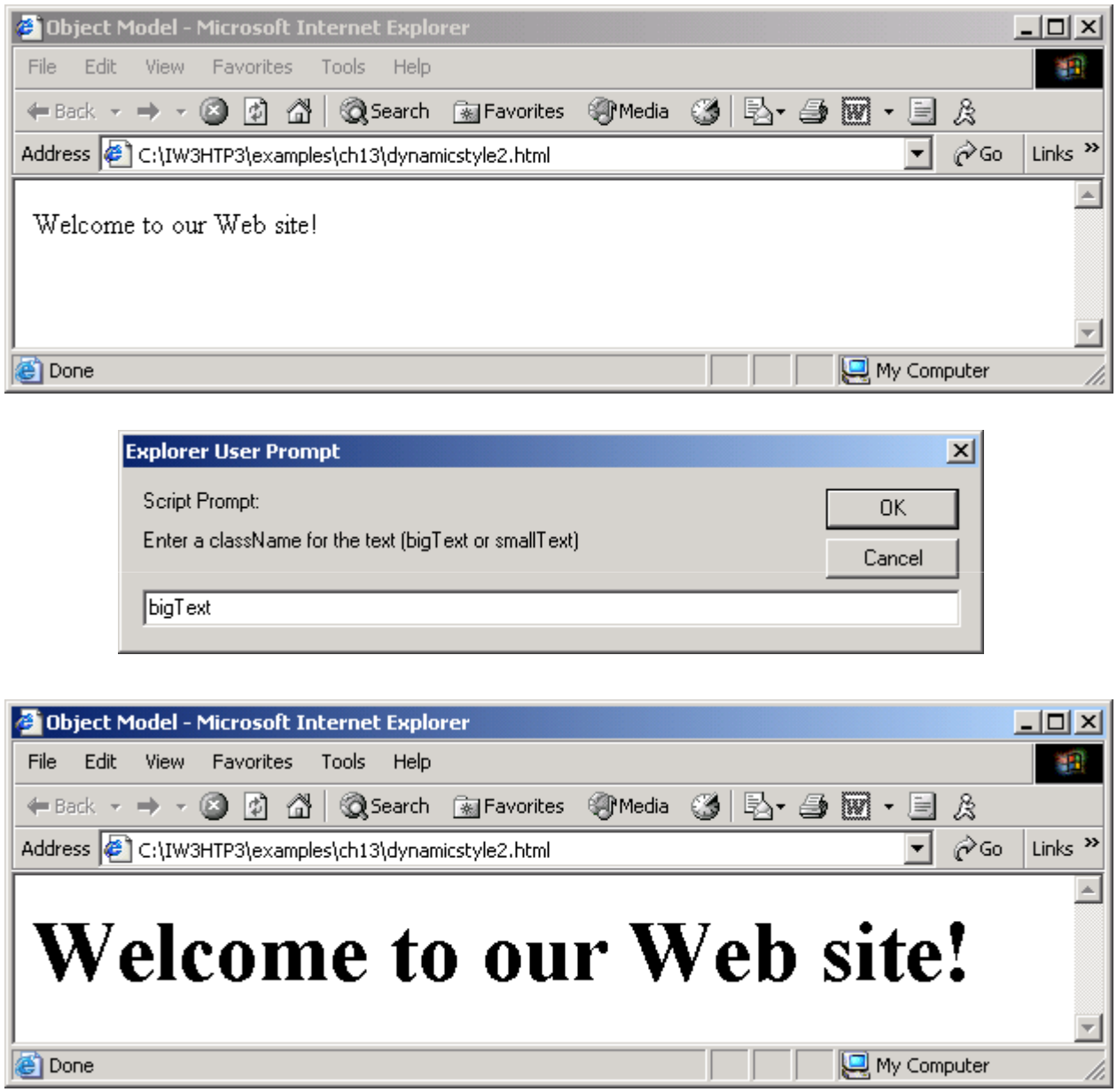
dynamicstyle2.html
(1 of 2)

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



Outline

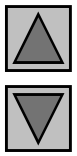
dynamicstyle2.html
(2 of 2)


Object Model - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address C:\IW3HTP3\examples\ch13\dynamicstyle2.html
Go Links
Welcome to our Web site!
Done
My Computer
Explorer User Prompt
Script Prompt:
Enter a className for the text (bigText or smallText)
OK
Cancel
bigText
Object Model - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address C:\IW3HTP3\examples\ch13\dynamicstyle2.html
Go Links
Welcome to our Web site!
Done
My Computer

13.5 Dynamic Positioning

- XHTML elements can be positioned with scripting
  - Declare an element's CSS position property to be either `absolute` or `relative`
  - Move the element by manipulating any of the `top`, `left`, `right` or `bottom` CSS properties

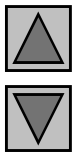
```
1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig. 13.6: dynamicposition.html -->
6  <!-- Dynamic Positioning             -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9     <head>
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
```



Outline

dynamicposition .html (1 of 3)

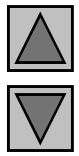
```
26          function run()
27          {
28             count += speed;
29
30             if ( ( count % 200 ) == 0 ) {
31                speed *= -1;
32                direction = !direction;
33
34                pText.style.color =
35                   ( speed < 0 ) ? "red" : "blue" ;
36                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
```



Outline

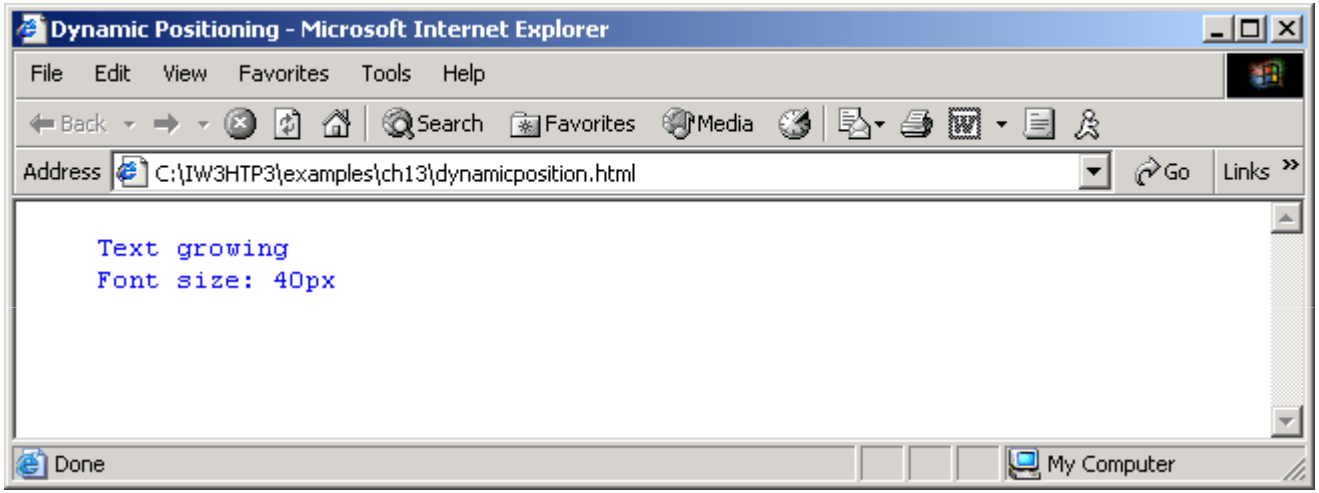
dynamicposition .html (2 of 3)

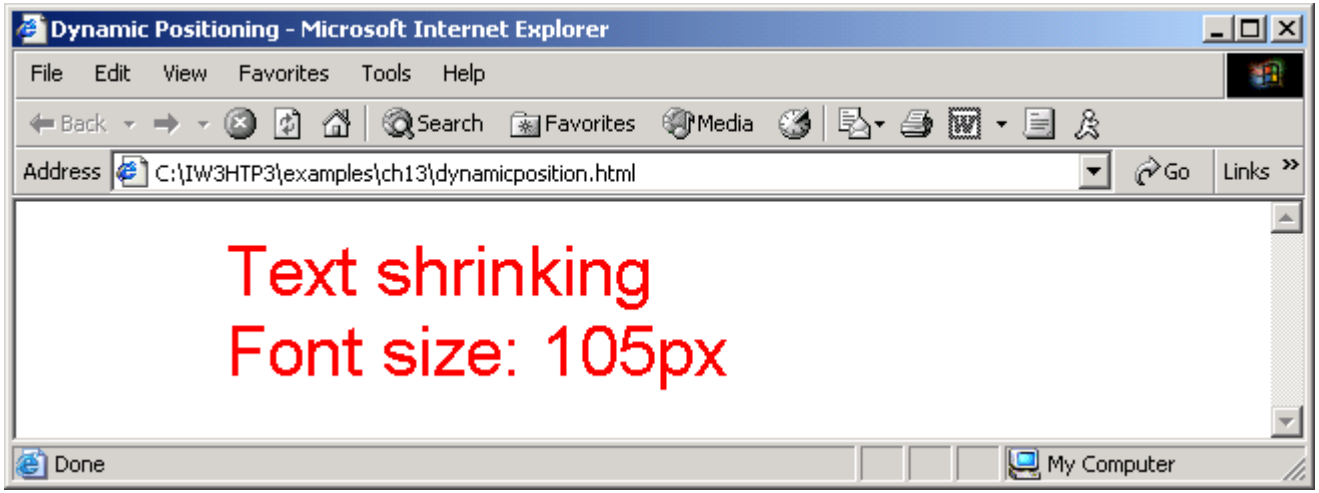
```
51    <body onload = "start()">
52          <p id = "pText" style = "position: absolute; left: 0;
53                                   font-family: serif; color: blue">
54          Welcome!</p>
55    </body>
56 </html>
```

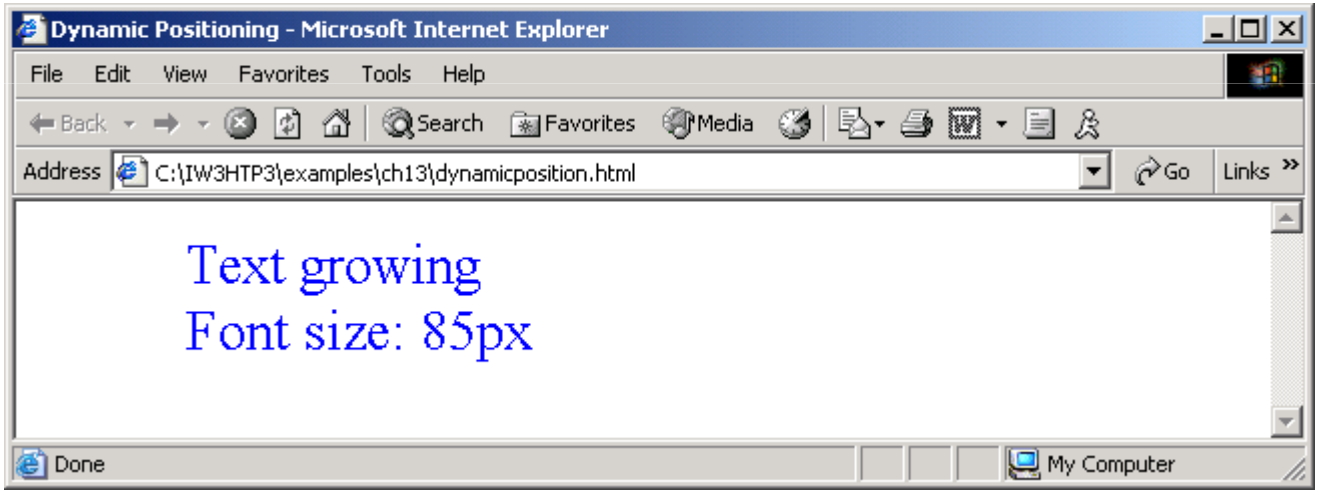


Outline

dynamicposition .html (3 of 3)



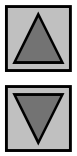

Dynamic Positioning - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address C:\IW3HTP3\examples\ch13\dynamicposition.html
Go Links
Text shrinking
Font size: 105px
Done
My Computer


Dynamic Positioning - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address C:\IW3HTP3\examples\ch13\dynamicposition.html
Go Links
Text growing
Font size: 85px
Done
My Computer

13.6 Using the frames Collection

- Referencing elements and objects in different frames by using the `frames` collection

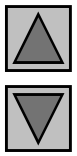
```
1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
3     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
4
5  <!-- Fig. 13.7: index.html       -->
6  <!-- Using the frames collection -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9     <head>
10       <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>
```



Outline

index.html
(1 of 1)

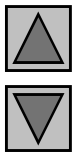
```
1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig. 13.8: top.html   -->
6  <!-- Cross-frame scripting -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9     <head>
10       <title>The 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
```



Outline

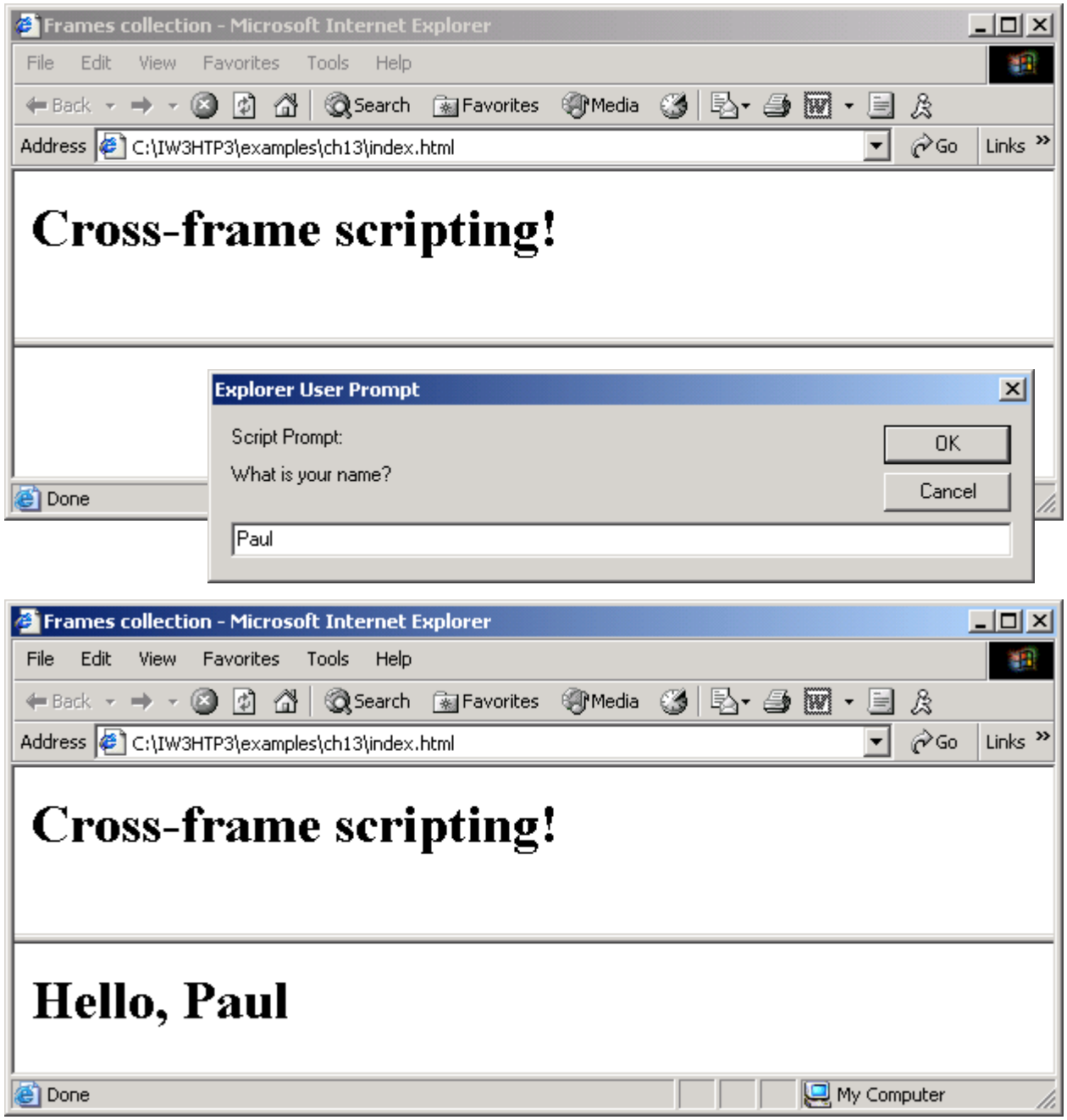
top.html
(1 of 2)

```
24     <body onload = "start()">
25        <h1>Cross-frame scripting!</h1>
26     </body>
27  </html>
```



Outline

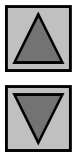
top.html
(2 of 2)


Frames collection - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address C:\IW3HTP3\examples\ch13\index.html
Go Links
Cross-frame scripting!
Explorer User Prompt
Script Prompt:
What is your name?
OK
Cancel
Paul
Done
Frames collection - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address C:\IW3HTP3\examples\ch13\index.html
Go Links
Cross-frame scripting!
Hello, Paul
Done
My Computer

13.7 `navigator` Object

- Netscape, Mozilla, Microsoft's Internet Explorer
  - Others as well
- Contains information about the Web browser
- Allows Web authors to determine what browser the user is using

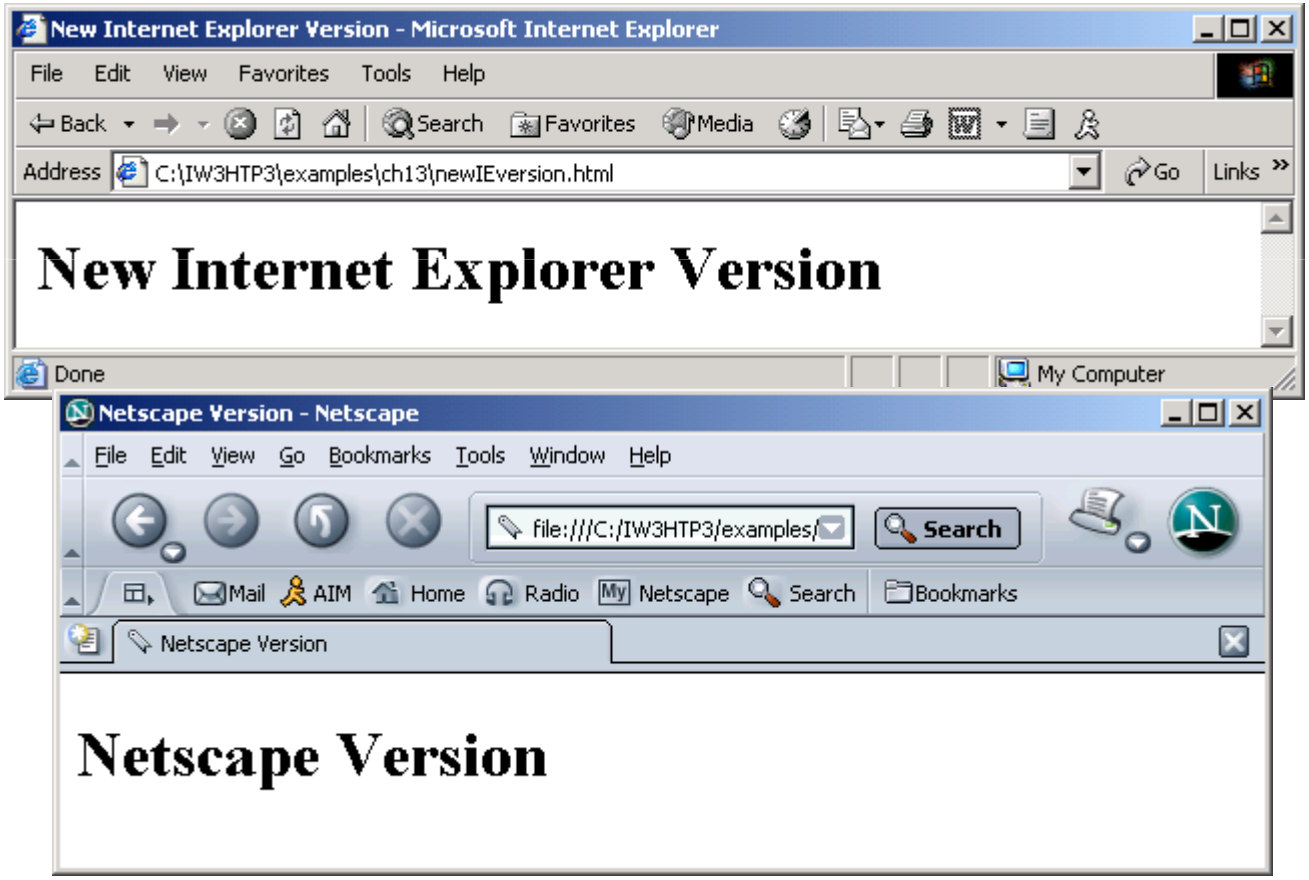
```
1  <?xml version = "1.0"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4
5  <!-- Fig 13.9: navigator.html   -->
6  <!-- Using the navigator object -->
7
8  <html xmlns = "http://www.w3.org/1999/xhtml">
9     <head>
10       <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          }
```



Outline

navigator.html
(1 of 2)

```
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>
```



Outline

navigator.html
(2 of 2)

13.8 Summary of the DHTML Object Model

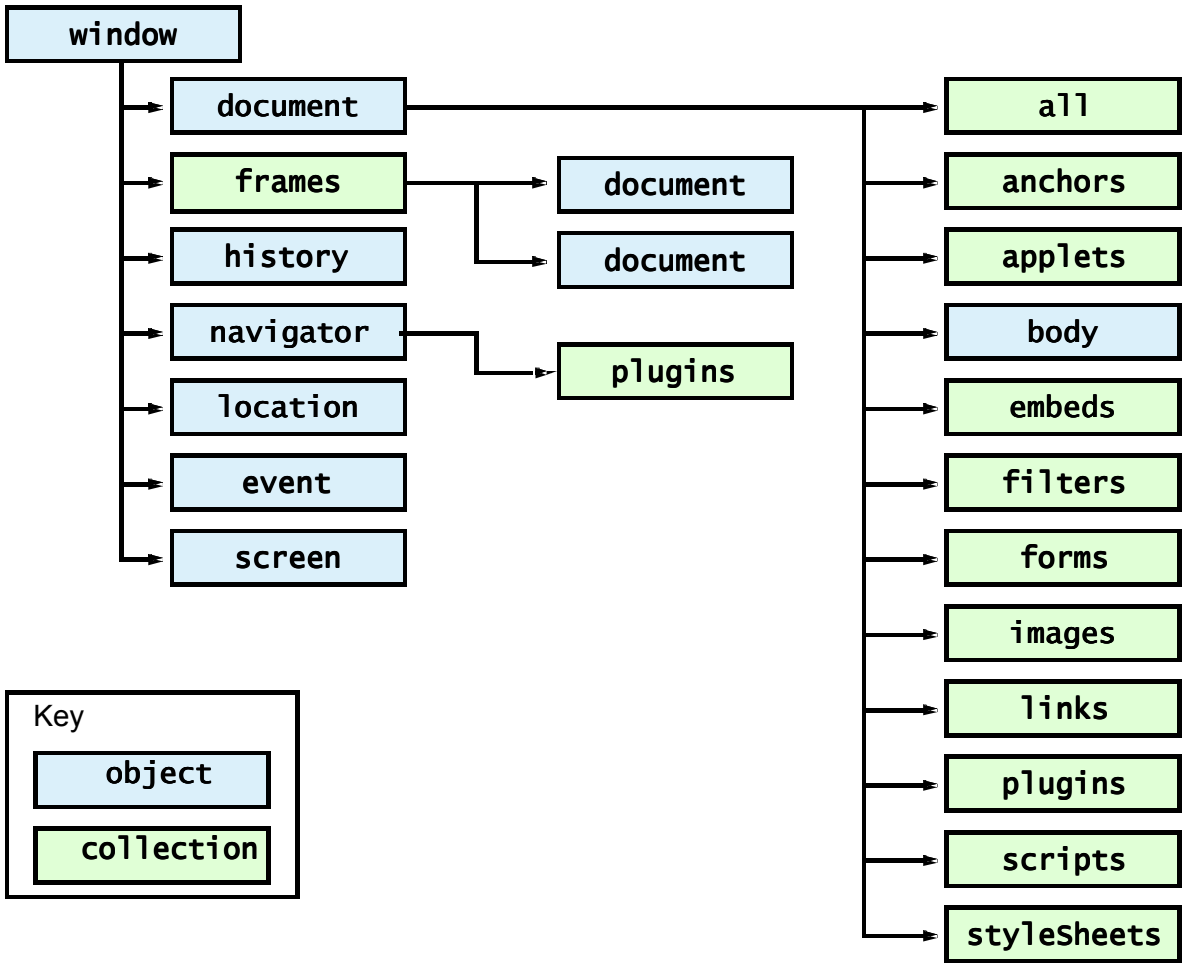


Fig. 13.10 DHTML Object Model.

13.8 Summary of the DHTML Object Model

Object or collection	Description
Objects	
`window`	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 the frame. Frames are considered to be subwindows in the browser.
`document`	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`	Provides access to the `body` element of an XHTML document.
`history`	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`	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`	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`	Can be used in an event handler to obtain information about the event that occurred (e.g., the mouse *x*-*y* coordinates during a mouse event).
`screen`	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 6 Object Model.

13.8 Summary of the DHTML Object Model

Object or collection	Description
Collections	
`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`	Collection contains all the 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`	Contains all the `applet` elements in the XHTML document. Currently, the most common `applet` elements are Java applets.
`embeds`	Contains all the `embed` elements in the XHTML document.
`forms`	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`	Contains `window` objects that represent each frame in the browser window. Each frame is treated as its own subwindow.
`images`	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`	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.

Fig. 13.11 Objects in the Internet Explorer 6 Object Model.

13.8 Summary of the DHTML Object Model

Object or collection	Description
`plugins`	Like the `embeds` collection, this collection contains all the `embed` elements in the XHTML document.
`scripts`	Contains all the `script` elements in the XHTML document.
`styleSheets`	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 6 Object Model.