

5.4 JavaScript Objects and Methods

JavaScript Objects and Properties

- Aggregate real world data types
- An object has properties and methods
 - Constructors for creating objects at run time
 - Accessing object properties
 - Accessing object methods

5.4.1 JavaScript Predefined Objects

images

arrays

strings

functions

documents

forms

frames

image

image.width

image.height

document.thisform.button

Creating Objects

```
var obj = new Object();
```

```
var today = new Date();
```

```
var point = new Object();
```

```
point.x = 10;
```

```
point.y = -10;
```

```
var rectangle = {upperLeftPoint: {x:1, y:1},  
                 lowerRightPoint: {x:4, y:4}};
```

Global objects

- Object Object
- Navigator object
- Number object
- String object
- Screen object
- event object
- Event object
- RegExp object – pattern matching in strings

Object

- A primitive JavaScript object type
- The constructor -- Object()
- Methods:
 - eval() ... JavaScript 1.1, 1.2
 - toString() ... JavaScript 1.1, 1.2
 - valueOf () ... JavaScript 1.1, 1.2
 - handleEvent() ... JavaScript 1.2
 - watch() ... JavaScript 1.2
 - unwatch() ... JavaScript 1.2

Navigator Object

- Getting the name of the Browser Language
- Getting the Name of the Platform
- Methods
 - preference()
- Properties

navigator.appCodeName
navigator.appName
navigator.appVersion
navigator.language
navigator.platform
navigator.userAgent

5.4.2 Number Object

- Implemented in the JavaScript 1.1 release (Navigator 3.0) and fix some minor things in JavaScript 1.2
- Properties
 - MAX_VALUE
 - MIN_VALUE
 - NaN
 - NEGATIVE_INFINITY
 - POSITIVE_INFINITY
 - prototype
- Methods
 - toString()
 - toString(radix))

5.4.3 document object

- Browser's document object
- Referencing to document object:
 - window.document
 - document
- It represents the HTML document currently displayed in the browser
- Has methods or functions
 - document.close() // Close an output stream
 - document.open(mimetype)// Begin a new document

- `write(value, ...)` // Append data to a document
 - `writeln(value, ...)` // Append data and a new line to a document
- Has attributes, properties, or variables
 - `document.bgColor` // The document background color
 - `document.alinkColor` // The color of active links
 - `document.vlinkColor` // The color of visited links
 - `document.linkColor` // The color of unfollowed links
 - `document.lastModified` // The modification date of a document
 - `document.URL` // The URL of the current document
 - `document.referrer` // The URL of the linked-from document
 - `document.title` // The title of the current document
 - `document.cookie`
 - `document.fgColor` // the default text color
 - `document.form[]`
 - `document.images[]`
- Pass argument to a method within the pair of parentheses

`document.write()`

- `write()` method
- No new line character is issued
- Examples:
 - `document.write("FONT COLOR = 'red'><H1>ECET ")`
 - `document.write("string for display")`
 - `document.write("Hello World!
")`
 - `document.write("Hello World! \r")`
 - `document.writeln("Hello World!")`

`document.writeln()`

- `writeln()` method for writing text
- A new line character is added
- Examples

- `document.writeln(" Department Welcomes You!</H1>")`

The document object

Synopsis

`window.document`
`document`

Properties

`alinkColor`
The color of activated links.

`bgColor`
The document background color

`fgColor`
The default text color.

`linkColor`
The color of unfollowed links.

`vlinkColor`
The color of visited links.

`lastModifies`
The modification date of a document.

`location`
The URL of the current document.

`title`
The title of the document.

`URL`
The Uniform Resource Locator of the current document.

`referrer`
The URL of the linked-from document.

`cookie`
The cookie(s) of the document.

`domain`
The security domain of the document

`form[]`
The forms in a document

`images[]`
The images embedded in a document.

`link[]`
The link objects in a document

`anchors[]`
The Anchors array in a document.

`applets[]`
The applet array in a document.

`plugins[]`
The object embed in a document.

`embeds[]`
The objects embedded in a document.

Navigator 4 and above Properties

`classes`
Define style classes.

`height`
The height of a document.

`width`
The width of a document.

`ids`
Define styles for individual tags.

`layers`
The layers contained in a document

`tags`
Define styles for HTML tags

Internet Explorer 4 and above Properties

`activeElement`

The input element that has the focus.

`charset`

The character set in use.

`defaultCharset`

The default character set of a document

`expando`

Disallow new property creation.

`parentWindow`

The window of a document

`readyState`

The loading status of a document.

`all[]`

All HTML elements in a document.

Methods

`close()`

Close an output stream.

`open(mimitype)`

Create a new document

`write(datavalue,...)`

Append data to a document.

`writeln(datavalue)`

Append a document and a new line to a document.

Methods for Netscape 4 and above

`captureEvents(eventmask)`
Specify the event types to be captures

`releaseEvents(eventmask)`
Stop capturing events

`routeEvents(event)`
Pass a captured event to the next handler

`getSelection()`
Return selected text

Methods for IE 4 and above

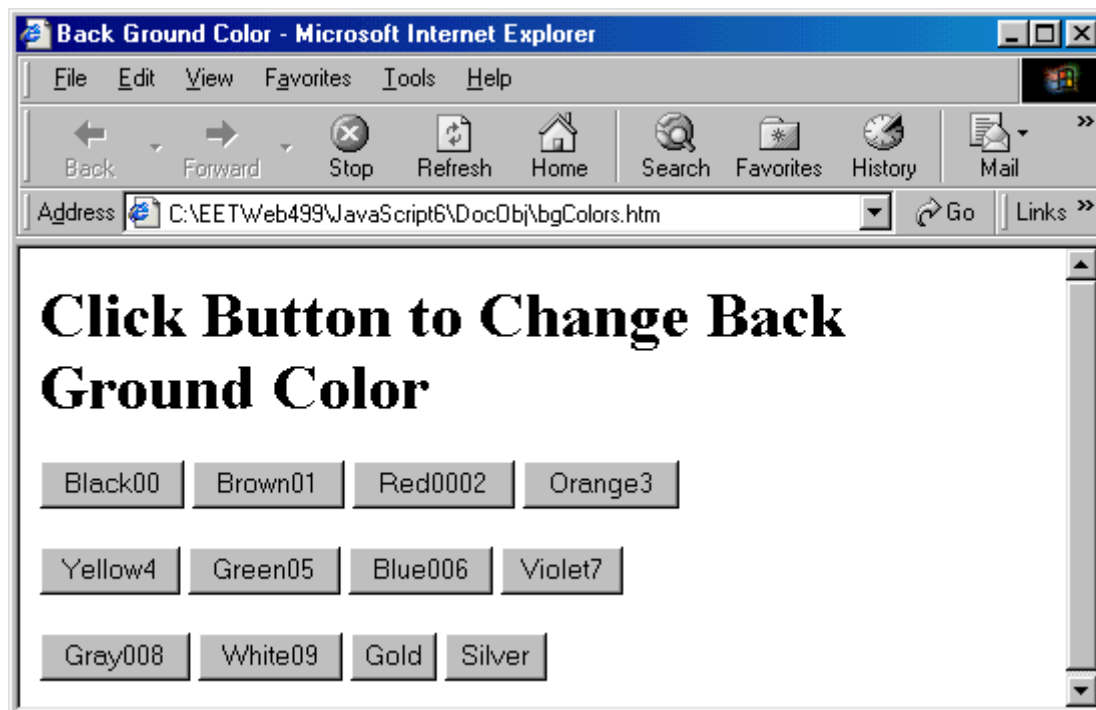
`elementFromPoint(x, y)`
Determine which HTML element is at a given point.

Example 5-13: Change background color properties through document object.

```

<HTML>
<!-- bgolor.htm
-->
<HEAD>
<TITLE>Back Ground Color</TITLE></HEAD>
<BODY>
<H1>Click Button to Change Back Ground Color</H1>
<FORM>
<P>
<INPUT TYPE="BUTTON" NAME="blcak" VALUE="Black00" ONCLICK='document.bgColor="black"'>
<INPUT TYPE="BUTTON" NAME="brown" VALUE="Brown01" ONCLICK='document.bgColor="brown"'>
<INPUT TYPE="BUTTON" NAME="red" VALUE="Red0002" ONCLICK='document.bgColor="red"'>
<INPUT TYPE="BUTTON" NAME="orange" VALUE="Orange3" ONCLICK='document.bgColor="orange"'>
</P>
<P>
<INPUT TYPE="BUTTON" NAME="yellow" VALUE="Yellow4" ONCLICK='document.bgColor="yellow"'>
<INPUT TYPE="BUTTON" NAME="green" VALUE="Green05" ONCLICK='document.bgColor="green"'>
<INPUT TYPE="BUTTON" NAME="blue" VALUE="Blue006" ONCLICK='document.bgColor="blue"'>
<INPUT TYPE="BUTTON" NAME="violet" VALUE="Violet7" ONCLICK='document.bgColor="violet"'>
</P>
<P>
<INPUT TYPE="BUTTON" NAME="gray" VALUE="Gray008" ONCLICK='document.bgColor="gray"'>
<INPUT TYPE="BUTTON" NAME="white" VALUE="White09" ONCLICK='document.bgColor="white"'>
<INPUT TYPE="BUTTON" NAME="gold" VALUE="Gold" ONCLICK='document.bgColor="gold"'>
<INPUT TYPE="BUTTON" NAME="silver" VALUE="Silver" ONCLICK='document.bgColor="silver"'>
</P>
</FORM>
</BODY>
</HTML>

```



5.4.4 Window Objects

Basic Window Operations

- Opening a window
- Scrolling a window
- Loading a content into a window
- Updating one window from another
- Creating new pages with JavaScript
- Closing a window
- Creating a control panel
- Positioning a window on the screen
- Displaying an alert when a window is loaded

Properties and Methods of the window() object

- Create another new window outside the browser
- Referencing to a window
 - self
 - window
 - window.frame[n]
- window properties
 - self // the window itself
 - window // the window itself
 - window.document // document of the window
 - window.name // name of the window
 - window.location // URL of the window
 - window.status // specify a line of status message
 - window.history // the history of the window
 - window.length // the number of frames in the window
 - window.screen // information about the screen
 - window.frames[] // List of frames within a window
- window methods
 - alert(message) // Display message through dialog box
 - prompt(message, defaultValue) // Receive input through dialog box

- `confirm(question)` // Ask to confirm a yes-no question
- `close()` // close a browser window
- `open(url, name, features, replace)` // open a new window
- `setInterval(code, interval)` // Repeatedly execute the code at a specific time interval
- `setTimeout(code, delay)` // Defer execution of code
- window event handlers
 - `onblur` // invoked when the window loses keyboard focus
 - `ondragdrop` // invoked when the user drops items in the window
 - `onerror` // invoked when Javascript
 - `onfocus` // invoked when window is given a focus
 - `onload` // invoked when browser completed the loading
 - `onmove` // invoked when window is moved
 - `onresize` // invoked when window is resized
 - `onunload` // invoked when the browser leaves a page

`window.alert()`

- `alert()` method for creating a dialog box
- OK button
- `window.alert("Thank you!\nTry Again!")`

`window.prompt()`

- `prompt()` is a dialog control
- Examples:
 - `firstNum = window.prompt("Enter the first integer number", "0");`
 - `secondNum = window.prompt("Enter the second integer number", "0");`

`window.open()`

- open a new window

```
newWindow1 = window.open(URL, "nameOfWindow", "toolbar=no, width=350, height=400, status=no, scrollbar=yes, resize=no, menubar=no");
```

```
newWindow2 = window.open('bulbon.gif', 'bulbWin', 'width=330, height=250,
scrollbars=yes')
```

```
newWindow3 = window.open('book.html', 'bookWin', 'width=330, height=250,
scrollbars=yes')
```

```
window.confirm()
```

- Display a message in a dialog box
- Click "OK" button to return a true value
- Click "Cancel" button to return a false value
- `confirm("message")`

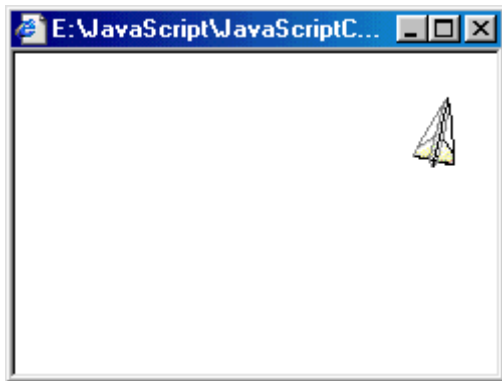
A basic window creation example

```
<HTML>
<!-- newwin1.htm -->
<HEAD>
<TITLE>Open a New Window </TITLE>
</HEAD>

<BODY>
<SCRIPT LANGUAGE="Javascript">
window.open(URL, "thisWindow", "toolbar=no, width=350, height=400,
status=no, scrollbars=yes, resize=no, menubar=no")
</SCRIPT>
</BODY>
</HTML>
```

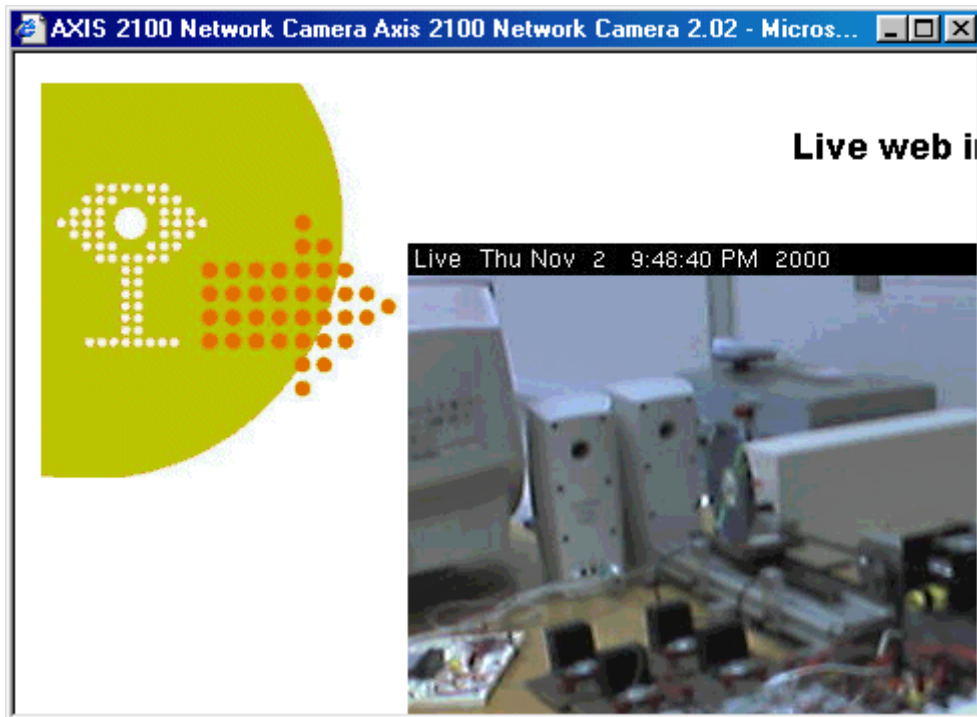
Example 5-14: Create a new window for displaying a animated airplane.gif graphics file.

```
HTML>
<!-- newplane.htm -->
<HEAD>
<TITLE>Open a New Window </TITLE>
<SCRIPT LANGUAGE="Javascript">
window.open("airplane.gif", "SUNWindow", "toolbar=no, width=350,
height=400, status=no, scrollbars=yes, resize=no, menubar=no")
</SCRIPT>
</HEAD>
<BODY>
</BODY>
</HTML>
```



Example 5-15: Create a new window for showing web camera. The newly window will stay open even if the browser is closed.

```
<HTML>
<!-- newlivecam.htm -->
<HEAD>
<TITLE>Open a New Window </TITLE>
<SCRIPT LANGUAGE="Javascript">
// window.open("http://149.164.36.19", "LiveCam", "toolbar=no,
width=780, height=600, status=yes, scrollbras=yes, resize=yes,
menubar=yes")
//window.open("http://149.164.36.19", "LiveCam", "width=320,
height=240")
window.open("http://149.164.36.19", "LiveCam", "width=480, height=330")
</SCRIPT>
</HEAD>
<BODY>
</BODY>
</HTML>
```



Example 5-16: Create a new window for Web camera, and close the newly created window. Javascript with functions are considered in the program.

```
<HTML>
<!-- Webcam.htm -->
<HEAD>
<TITLE>Open a New Window </TITLE>
<SCRIPT LANGUAGE="Javascript">
function webCamera(URL){
    newWindow = window.open(URL, "LiveCam", "toolbar=no, width=780,
height=600, status=no, scrollbars=yes, resize=no, menubar=no")
}
function closeWindow() {if(newWindow && !newWindow.closed)
    newWindow.close(); }
</SCRIPT>
</HEAD>
<BODY>
<H3> <A HREF="javascript:webCamera('http://149.164.36.19')"> Click Here
To See WebCam <A>&nbsp; &nbsp; <BR> <BR>
    <A HREF="javascript:closeWindow()"> Close the Window <A>
</H3>
</BODY>
</HTML>
```

