30

Built-in Objects

JAVASCRIPT

31

String

- Khởi tạo: myString = new String ("Hello");
- Thuộc tính: length (chiều dài kí tự),

  ví dụ: alert(myString.length) //5
- Phương thức:

Method	Purpose
`anchor(name)`	Creates an anchor element (an `<a>` element with a `name` or `id` attribute rather than an `href` attribute).
`big()`	Displays text as if in a `<big>` element.
`bold()`	Displays text as if in a `<bold>` element.
`charAt(index)`	Returns the character at a specified position (for example, if you have a string that says "banana" and your method reads `charAt(2)` then you will end up with the letter `n` — remember that indexes start at 0).
`fixed()`	Displays text as if in a `<tt>` element.

String (tt)

32

- Phương thức (tt)

`fontcolor(color)`	Displays text as if in a `<font>` element with a `color` attribute.
`fontsize(fontsize)`	Displays text as if in a `<font>` element with a `size` attribute.
`indexOf(searchValue, [fromindex])`	Returns the position of the first occurrence of the specified string `searchValue` inside another string. For example, if you have the word "banana" as your string, and you want to find the first occurrence of the letter `n`, you use `indexOf(n)`. If the `fromIndex` argument is used, the search will begin at that index. For example, you might want to start after the fourth character. The method returns `-1` if the string being searched for never occurs.

String (tt)

33

- Phương thức (tt)

Method	Purpose
`italics()`	Displays text as if in an `<i>` element.
`lastIndexOf(searchValue, [fromIndex])`	Same as `indexOf()` method, but runs from right to left.
`link(targetURL)`	Creates a link in the document.
`small()`	Displays text as if in a `<small>` element.
`strike()`	Displays text as if in a `<strike>` element.
`sub()`	Displays text as if in a `<sub>` element.
`substr(start}, [length])`	Returns the specified characters. 14,7 returns 7 characters, from the 14th character (starts at 0). Note that this works only in IE4 and Netscape 4 and later versions.
`substring(startPosition, endPosition)`	Returns the specified characters between the start and end index points. 7,14 returns all characters from the 7th up to but not including the 14th (starts at 0).
`sup()`	Displays text as if in a `<sup>` element.
`toLowerCase()`	Converts a string to lowercase.
`toUpperCase()`	Converts a string to uppercase.

Date

34

- Khởi tạo đối tượng date

  new Date()
  new Date(milliseconds)
  new Date(dateString)
  new Date(year, month, day, hours, minutes, seconds, milliseconds)

```
<script>
var d = new Date();
document.getElementById("demo").innerHTML = d;
</script>
```

```
<script>
var d = new Date(99, 5, 24);
document.getElementById("demo").innerHTML = d;
</script>
```

Date - methods

```
var current = new Date();
Day = current.getDay();
```

Method	Description
getDate() getUTCDate()	Returns a number from 1 to 31 representing the day of the month in local time or UTC.
getDay() getUTCDay()	Returns a number from 0 (Sunday) to 6 (Saturday) representing the day of the week in local time or UTC.
getFullYear() getUTCFullYear()	Returns the year as a four-digit number in local time or UTC.
getHours() getUTCHours()	Returns a number from 0 to 23 representing hours since midnight in local time or UTC.
getMilliseconds() getUTCMilliSeconds()	Returns a number from 0 to 999 representing the number of milliseconds in local time or UTC, respectively. The time is stored in hours, minutes, seconds and milliseconds.
getMinutes() getUTCMinutes()	Returns a number from 0 to 59 representing the minutes for the time in local time or UTC.
getMonth() getUTCMonth()	Returns a number from 0 (January) to 11 (December) representing the month in local time or UTC.
getSeconds() getUTCSeconds()	Returns a number from 0 to 59 representing the seconds for the time in local time or UTC.
getTime()	Returns the number of milliseconds between January 1, 1970, and the time in the Date object.
getTimezoneOffset()	Returns the difference in minutes between the current time on the local computer and UTC (Coordinated Universal Time).

Date - methods

Method	Description
`setMinutes( m, s, ms )` `setUTCMinutes( m, s, ms )`	Sets the minute in local time or UTC. The second and third arguments, representing the seconds and milliseconds, are optional. If an optional argument is not specified, the current value in the `Date` object is used.
`setMonth( m, d )` `setUTCMonth( m, d )`	Sets the month in local time or UTC. The second argument, representing the date, is optional. If the optional argument is not specified, the current date value in the `Date` object is used.
`setSeconds( s, ms )` `setUTCSeconds( s, ms )`	Sets the second in local time or UTC. The second argument, representing the milliseconds, is optional. If this argument is not specified, the current millisecond value in the `Date` object is used.
`setTime( ms )`	Sets the time based on its argument—the number of elapsed milliseconds since January 1, 1970.
`setDate( val )` `setUTCDate( val )`	Sets the day of the month (1 to 31) in local time or UTC.
`setFullYear( y, m, d )` `setUTCFullYear( y, m, d )`	Sets the year in local time or UTC. The second and third arguments representing the month and the date are optional. If an optional argument is not specified, the current value in the `Date` object is used.
`setHours( h, m, s, ms )` `setUTCHours( h, m, s, ms )`	Sets the hour in local time or UTC. The second, third and fourth arguments, representing the minutes, seconds and milliseconds, are optional. If an optional argument is not specified, the current value in the `Date` object is used.
`setMilliSeconds( ms )` `setUTCMilliseconds( ms )`	Sets the number of milliseconds in local time or UTC.

Date - methods

37

`toLocaleString()`	Returns a string representation of the date and time in a form specific to the computer's locale. For example, September 13, 2007, at 3:42:22 PM is represented as *09/13/07 15:47:22* in the United States and *13/09/07 15:47:22* in Europe.
`toUTCString()`	Returns a string representation of the date and time in the form: *15 Sep 2007 15:47:22 UTC*
`toString()`	Returns a string representation of the date and time in a form specific to the locale of the computer (*Mon Sep 17 15:47:22 EDT 2007* in the United States).
`valueOf()`	The time in number of milliseconds since midnight, January 1, 1970. (Same as `getTime`.)

Date - methods

38

Ví dụ:

var current = new Date();

```
document.writeln( "getDate: " + current.getDate() +
   "<br />getDay: " + current.getDay() +
   "<br />getMonth: " + current.getMonth() +
   "<br />getFullYear: " + current.getFullYear() +
   "<br />getTime: " + current.getTime() +
   "<br />getHours: " + current.getHours() +
   "<br />getMinutes: " + current.getMinutes() +
   "<br />getSeconds: " + current.getSeconds() +
   "<br />getMilliseconds: " + current.getMilliseconds() +
   "<br />getTimezoneOffset: " + current.getTimezoneOffset() );
```

Bài tập: Date

39

1. Viết function hiển thị ngày tháng năm hiện tại, kiểm tra xem năm hiện tại có phải là năm nhuận hay không
2. Viết function hiển thị các ngày và thứ trong tuần hiện tại
3. Viết function hiển thị các ngày trong tháng hiện tại, dạng bảng có 7 cột, cột đầu là thứ 2.
4. Viết function hiển thị năm âm lịch của 1 năm dương lịch
5. Viết function hiển ngày tháng năm, giờ phút giây của các thanh phố trên thế giới: New York, London, Tokyo, Paris, Hà Nội

Math

40

document.writeln(Math.sqrt(900.0));

Method	Description	Examples

`abs( x )`	absolute value of x	`abs( 7.2 )` is `7.2`
		`abs( 0.0 )` is `0.0`
		`abs( -5.6 )` is `5.6`

`ceil( x )`	rounds x to the smallest integer not less than x	`ceil( 9.2 )` is `10.0` `ceil( -9.8 )` is `-9.0`
`cos( x )`	trigonometric cosine of x (x in radians)	`cos( 0.0 )` is `1.0`
`exp( x )`	exponential method e^x	`exp( 1.0 )` is `2.71828` `exp( 2.0 )` is `7.38906`

Math

41

Method	Description	Examples
`floor( x )`	rounds x to the largest integer not greater than x	`floor( 9.2 )` is `9.0` `floor( -9.8 )` is `-10.0`
`log( x )`	natural logarithm of x (base *e*)	`log( 2.718282 )` is `1.0` `log( 7.389056 )` is `2.0`
`max( x, y )`	larger value of x and y	`max( 2.3, 12.7 )` is `12.7` `max( -2.3, -12.7 )` is `-2.3`
`min( x, y )`	smaller value of x and y	`min( 2.3, 12.7 )` is `2.3` `min( -2.3, -12.7 )` is `-12.7`
`pow( x, y )`	x raised to power y (x^y)	`pow( 2.0, 7.0 )` is `128.0` `pow( 9.0, .5 )` is `3.0`
`round( x )`	rounds x to the closest integer	`round( 9.75 )` is `10` `round( 9.25 )` is `9`
`sin( x )`	trigonometric sine of x (x in radians)	`sin( 0.0 )` is `0.0`
`sqrt( x )`	square root of x	`sqrt( 900.0 )` is `30.0` `sqrt( 9.0 )` is `3.0`
`tan( x )`	trigonometric tangent of x (x in radians)	`tan( 0.0 )` is `0.0`

Math

42

Constant	Description	Value
`Math.E`	Base of a natural logarithm (*e*).	Approximately 2.718
`Math.LN2`	Natural logarithm of 2	Approximately 0.693
`Math.LN10`	Natural logarithm of 10	Approximately 2.302
`Math.LOG2E`	Base 2 logarithm of *e*	Approximately 1.442
`Math.LOG10E`	Base 10 logarithm of *e*	Approximately 0.434
`Math.PI`	π—the ratio of a circle's circumference to its diameter	Approximately 3.141592653589793
`Math.SQRT1_2`	Square root of 0.5	Approximately 0.707
`Math.SQRT2`	Square root of 2.0	Approximately 1.414

Array

43

1. Khởi tạo đối tượng array: var a = new array(number)
2. Khai báo và khởi tạo items:
   var colors = new array ("blue", "black", "gray")
3. Khởi tạo danh sách items của mảng: var a = [1,2,3,4,5]

Array

- Ví dụ

```
<title>Sum the Elements of an Array</title>

<script type = "text/javascript">
   <!--
   var theArray = [ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 ];
   var total1 = 0, total2 = 0;

for ( var i = 0; i < theArray.length; i++ )
   total1 += theArray[ i ];

document.writeln( "Total using subscripts: " + total1 );
```

Array

- Thuộc tính:

TT	Thuộc tính	Ý nghĩa	Ví dụ
1	**Length**	Số lượng phần tử	a.length

- Các methods: xem chi tiết tại: https://www.w3schools.com/jsref/jsref_obj_array.asp

Mảng nhiều chiều

46

- Arrays of One-Dimensional Arrays
- Khai báo và khởi tạo như mảng 01 chiều

  var b = [[1, 2], [3, 4]];

```
var array1 = [ [ 1, 2, 3 ], // first row
               [ 4, 5, 6 ] ]; // second row
var array2 = [ [ 1, 2 ], // first row
               [ 3 ], // second row
               [ 4, 5, 6 ] ]; // third row
```

- Khai báo với new:

```
var b;
b = new Array( 2 );
b[ 0 ] = new Array( 5 );
b[ 1 ] = new Array( 3 );
```

Array

47

- Ví dụ:

Values in array1 by row

1 2 3

4 5 6

Values in array2 by row

1 2

3

4 5 6

```
<script type = "text/javascript">
   <!--
   var array1 = [ [ 1, 2, 3 ], // first row
                  [ 4, 5, 6 ] ]; // second row
   var array2 = [ [ 1, 2 ], // first row
                  [ 3 ], // second row
                  [ 4, 5, 6 ] ]; // third row

   outputArray( "Values in array1 by row", array1 );
   outputArray( "Values in array2 by row", array2 );

   function outputArray( heading, theArray )
   {
      document.writeln( "<h2>" + heading + "</h2><pre>" );

      // iterates through the set of one-dimensional arrays
      for ( var i in theArray )
      {
         // iterates through the elements of each one-dimensional
         // array
         for ( var j in theArray[ i ] )
            document.write( theArray[ i ][ j ] + " " );

         document.writeln( "<br />" );
      } // end for

      document.writeln( "</pre>" );
   } // end function outputArray
   // -->
</script>
```

Window

- Quản lý cửa sổ trình duyệt

Method or property	Description
`open(` *url*, *name*, *options* `)`	Creates a new window with the URL of the window set to *url*, the name set to *name* to refer to it in the script, and the visible features set by the string passed in as *option*.
`prompt(` *prompt*, *default* `)`	Displays a dialog box asking the user for input. The text of the dialog is *prompt*, and the default value is set to *default*.
`close()`	Closes the current window and deletes its object from memory.
`focus()`	This method gives focus to the window (i.e., puts the window in the foreground, on top of any other open browser windows).
`blur()`	This method takes focus away from the window (i.e., puts the window in the background).
`window.document`	This property contains the `document` object representing the document currently inside the window.
`window.closed`	This property contains a boolean value that is set to true if the window is closed, and false if it is not.