

DHTML

JAVA
SCRIPT

LẬP TRÌNH
WEB

PHP

Chương 5 JAVASCRIPT

Trình bày: Nguyễn Phú Trường

Giới thiệu về JavaScript

Biến, kiểu dữ liệu, phép toán

Lệnh điều khiển

Popup

Sử dụng các đối tượng

Giới thiệu về JavaScript

Biến, kiểu dữ liệu, phép toán

Lệnh điều khiển

Popup

Sử dụng các đối tượng

JAVASCRIPT LÀ GÌ?



1

Được thiết kế cùng với HTML tạo trang Web sinh động

Là ngôn ngữ script, hướng đối tượng, chứa các dòng lệnh thực thi được

2

Được viết trực tiếp vào trang HTML

Là ngôn ngữ thông dịch

3

Khác với Java

JAVASCRIPT DÙNG ĐỂ LÀM GÌ?



VÍ DỤ 1 VỀ JAVASCRIPT



```
<html>
<body>

<h1>My First Web Page</h1>

<script type="text/javascript">
    document.write("<p>" + Date() + "</p>");
</script>

</body>
</html>
```

VÍ DỤ 2 VỀ JAVASCRIPT



```
<html>
<body>
```

```
<h1>My First Web Page</h1>
```

```
<p id="demo"></p>
```

```
<script type="text/javascript">
document.getElementById("demo").innerHTML=Date();
</script>
```

```
</body>
</html>
```

- ❖ Để chèn một JavaScript vào trang HTML, dùng thẻ `<script>`.
- ❖ Trong thẻ `<script>` dùng thuộc tính kiểu để định nghĩa ngôn ngữ kịch bản.
- ❖ Cặp thẻ `<script>` và `</script>` xác định nơi JavaScript bắt đầu và kết thúc
- ❖ Các dòng giữa `<script>` và `</script>` chứa JavaScript và được thực thi bởi trình duyệt.



```
❖ <html>
  <body>
    <h1>My First Web Page</h1>

    <p id="demo">This is a paragraph.</p>

    <script type="text/javascript">
      ... some JavaScript code ...
    </script>

  </body>
</html>
```

- ❖ Các trình duyệt không hỗ trợ JavaScript, sẽ hiển thị JavaScript như nội dung trang.
- ❖ Để ngăn chặn chúng thực hiện điều này, và như một phần của chuẩn JavaScript, thẻ chú thích nên được dùng để “giấu” JavaScript.
- ❖ Chỉ thêm thẻ chú thích HTML `<!--` trước câu lệnh JavaScript đầu tiên, và `-->` (kết thúc chú thích) sau dòng lệnh JavaScript cuối

```
❖ <html>
  <body>
    <script type="text/javascript">
      <!--
      document.getElementById("demo").innerHTML=Date();
      //-->
    </script>
  </body>
</html>
```

❖ JavaScripts có thể được đặt trong phần <body> và <head> của một trang HTML.

```
<html>
<body> <h1>My First Web Page</h1>
<p id="demo"></p>
<script type="text/javascript">
    document.getElementById("demo").innerHTML=Date();
</script>
</body>
</html>
```

- ❖ JavaScripts trong trang HTML sẽ được thực thi khi trang nạp. Điều này không luôn luôn những gì chúng ta muốn.
- ❖ Đôi khi chúng ta thực thi JavaScript khi sự kiện-**event** xảy ra, như khi người dùng nhấp một nút. Đây là trường hợp chúng ta đặt script trong một hàm-**function**.
- ❖ Các sự kiện thường được dùng kết hợp với hàm (giống như gọi hàm khi sự kiện xảy ra).

```
<html> <head>
<script type="text/javascript">
function displayDate()
{
document.getElementById("demo").innerHTML=Date();
}
</script>
</head>
<body>
<h1>My First Web Page</h1>
<p id="demo"></p>
<button type="button" onclick="displayDate()">Display Date</button>
</body>
</html>
```

- ❖ JavaScript có thể được đặt trong các tập tin ngoài.
- ❖ Các tập tin ngoài thường chứa mã được dùng cho nhiều trang web.
- ❖ Các tập tin JavaScript ngoài có phần mở rộng .js.
- ❖ **Chú ý:** script ngoài không chứa thẻ `<script></script>` !
- ❖ Để dùng tập tin ngoài, chỉ tới tập tin .js trong thuộc tính "src" của thẻ `<script>`

```
<html>
<head>
    <script type="text/javascript" src="xxx.js"></script>
</head>
<body>
</body>
</html>
```



```
<html> <head></head>
```

```
<body>
```

```
<script type="text/javascript" src="xxx.js">
```

```
</script>
```

```
<p>
```

The actual script is in an external script file called "xxx.js".

```
</p>
```

```
</body></html>
```

Nội dung tập tin xxx.js

```
document.write("This text was written by an external script!")
```

Đặc điểm

- Chứa dữ liệu
- Phân biệt hoa và thường
- Phạm vi:
 - Toàn cục
 - Cục bộ

Khai báo

- `var strname = some value`
- Hoặc*
- `strname = some value`

Gán trị

- `var strname = "Hege"`
- Hoặc*
- `strname = "Hege"`

```
<html>
<body>

<script type="text/javascript">
var name = "Hege"
document.write(name)
document.write("<h1>" + name + "</h1>")
</script>

<p>This example declares a variable,
assigns a value to it, and then displays
the variable.</p>

<p>Then the variable is displayed one
more time, only this time as a
heading.</p>

</body>
</html>
```

Hege

Hege

This example declares a variable, assigns a value to it, and then displays the variable.

Then the variable is displayed one more time, only this time as a heading.

■ Kiểu dữ liệu

- Số nguyên : 10, -301, 1974, etc.
- Số thực và số chấm động : 13.5, 1.35E1
- Luận lý : true, false
- Chuỗi : “do thanh nghi”, “sinh nam 1974”, \b, \n, \r, \t, \\
var quote = "He read \"The Cremation of Sam McGee\" by R.W. Service."
document.write(quote)

Kết quả sẽ là :

He read "The Cremation of Sam McGee" by R.W. Service.

■ Kiểu dữ liệu

● Mảng :

```
myArray = new Array(10); foo = new Array(5);
```

```
myArray[0] = 56; myArray[9] = 44;
```

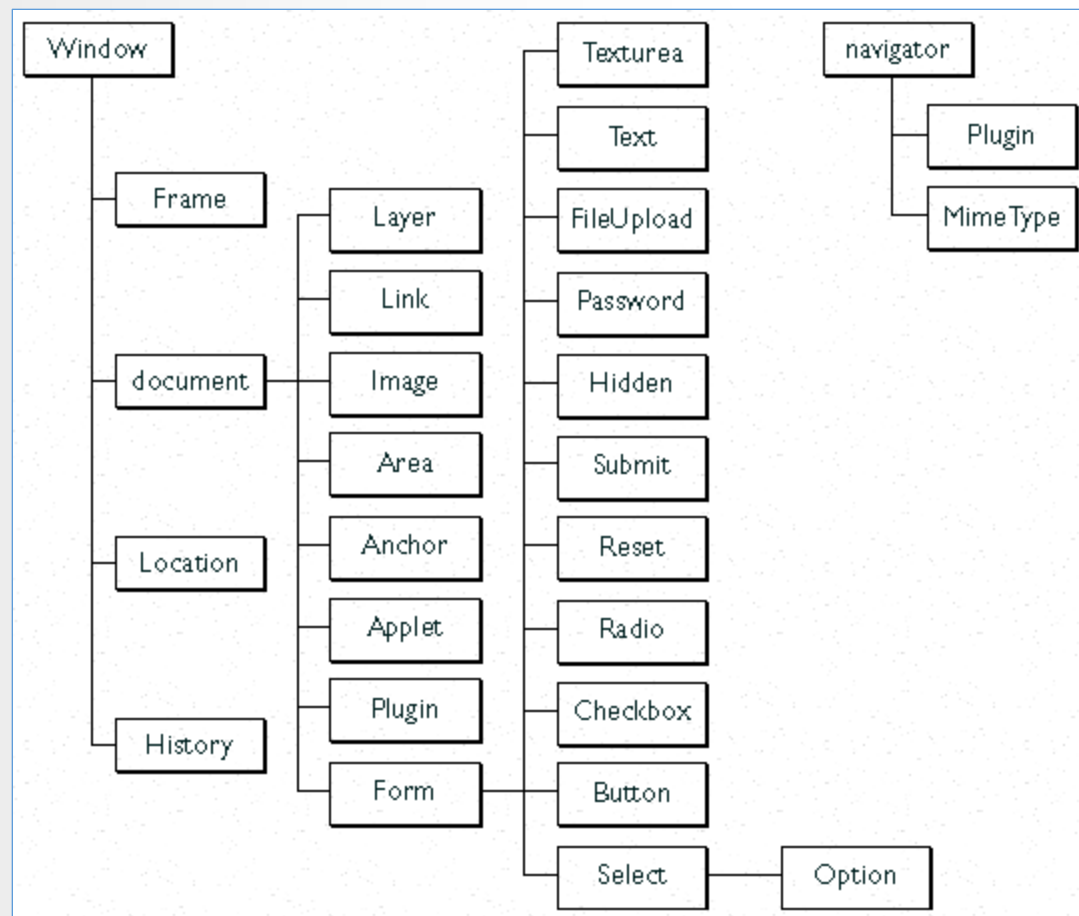
```
colors = new Array();
```

```
colors[99] = "midnightblue";
```

```
numberOfElements = myArray.length;
```

■ Kiểu dữ liệu

- Đối tượng :



■ Kiểu dữ liệu

- Được chuyển tự động :

```
var answer = 42
```

```
answer = "Thanks for all the fish..."
```

```
x = "The answer is " + 42
```

- Chuyển chuỗi sang số : `parseInt()`, `parseFloat()`

■ Phép toán

- Gán : =
- Phép tăng hoặc giảm 1 : ++, --
- Gán rút gọn : +=, -=, *=, /=, %=
- So sánh : ==, !=, <, <=, >, >=
- Phép tính số học : +, -, *, /, %
- Phép toán luận lý : &&, ||, !
- Phép : (cond) ? Expr1:Expr2

PHÉP TOÁN



Operator	Description	Example	Result
+	Addition	x=2 y=2 x+y	4
-	Subtraction	x=5 y=2 x-y	3
*	Multiplication	x=5 y=4 x*y	20
/	Division	15/5 5/2	3 2.5
%	Modulus (division remainder)	5%2 10%8 10%2	1 2 0
++	Increment	x=5 x++	x=6
--	Decrement	x=5 x--	x=4

PHÉP TOÁN



Operator	Example	Is The Same As
=	$x = y$	$x = y$
+=	$x += y$	$x = x + y$
-=	$x -= y$	$x = x - y$
*=	$x *= y$	$x = x * y$
/=	$x /= y$	$x = x / y$
%=	$x \% = y$	$x = x \% y$

Operator	Description	Example
<code>==</code>	is equal to	<code>5==8</code> returns false
<code>===</code>	is equal to (checks for both value and type)	<code>x=5</code> <code>y="5"</code> <code>x==y</code> returns true <code>x===y</code> returns false
<code>!=</code>	is not equal	<code>5!=8</code> returns true
<code>></code>	is greater than	<code>5>8</code> returns false
<code><</code>	is less than	<code>5<8</code> returns true
<code>>=</code>	is greater than or equal to	<code>5>=8</code> returns false
<code><=</code>	is less than or equal to	<code>5<=8</code> returns true

Operator	Description	Example
&&	and	<pre>x=6 y=3</pre> <p>(x < 10 && y > 1) returns true</p>
	or	<pre>x=6 y=3</pre> <p>(x==5 y==5) returns false</p>
!	not	<pre>x=6 y=3</pre> <p>!(x==y) returns true</p>

```
txt1="What a very"
txt2="nice day!"
txt3=txt1+txt2

txt1="What a very"
txt2="nice day!"
txt3=txt1+" "+txt2

txt1="What a very "
txt2="nice day!"
txt3=txt1+txt2
```

■ Định nghĩa

```
function functionname(var1,var2,...,varX)
{
some code
}
```

■ Ví dụ

```
function total(a,b)
```

```
{
```

```
x=a*b
```

```
return x
```

```
}
```

```
product=total(2,3)
```

■ Ví dụ

```
<html> <head>
<script type="text/javascript">
function displaymessage()
{
alert("Hello World!")
}
</script>
</head><body>
<form>
<input type="button" value="Click me!"
onclick="displaymessage()" >
</form>
</body>
```


ĐỊNH NGHĨA HÀM



```
<html>
<head>

<script type="text/javascript">
function myFunction()
{
return ("Hello, have a nice day!")
}
</script>

</head>
<body>

<script type="text/javascript">
document.write(myFunction())
</script>

<p>The script in the body section calls
a function.</p>

<p>The function returns a text.</p>

</body>
</html>
```

Hello, have a nice day!

The script in the body section calls a function.

The function returns a text.

ĐỊNH NGHĨA HÀM



```
<html>
<head>
<script type="text/javascript">
function total(a,b)
{
return a*b
}
</script>
</head>

<body>
<script type="text/javascript">
document.write(total(4,3))
</script>
<p>The script in the body section calls a
function with two parameters (4 and 3).</p>
<p>The function will return the product of
these two parameters.</p>
</body>
</html>
```

12

The script in the body section calls a function with two parameters (4 and 3).

The function will return the product of these two parameters.

Giới thiệu về JavaScript

Biến, kiểu dữ liệu, phép toán

Lệnh điều khiển

Popup

Sử dụng các đối tượng

■ Cú pháp

```
if (condition) {
    statements1
}
```

Hay

```
if (condition) {
    statements1
}
else {
    statements2
}
```

CẤU TRÚC IF-ELSE



```
<script type="text/javascript">
var d = new Date()
var time = d.getHours()

if (time < 10)
{
document.write("<b>Good morning</b>")
}
else
{
document.write("<b>Good day</b>")
}
</script>

<p>
This example demonstrates the If...Else
statement.
</p>

<p>
If the time on your browser is less than
10,
you will get a "Good morning" greeting.
Otherwise you will get a "Good day"
greeting.
```

Good day

This example demonstrates the If...Else statement.

If the time on your browser is less than 10, you will get a "Good morning" greeting. Otherwise you will get a "Good day" greeting.

CẤU TRÚC IF-ELSE



```
<html>
<body>

<script type="text/javascript">
var r=Math.random()
if (r>0.5)
{
document.write("<a
href='http://www.w3schools.com'>Learn Web
Development!</a>")
}
else
{
document.write("<a
href='http://www.refsnesdata.no'>Visit
Refsnes Data!</a>")
}
</script>

</body>
</html>
```

[Learn Web Development!](http://www.w3schools.com)

■ Cú pháp

```
switch(n) {
    case 1:
        execute code block 1
        break;
    case 2:
        execute code block 2
        break;
    default:
        execute code block n
}
```

CẤU TRÚC SWITCH-CASE



■ Ví dụ

```
<script type="text/javascript">
    var d = new Date()
    theDay=d.getDay()
    switch (theDay) {
        case 5:
            document.write("Finally Friday")
            break;
        case 6:
            document.write("Super Saturday");
            break;
        case 0:
            document.write("Sleepy Sunday");
            break
        default:
            document.write("I'm looking forward to this weekend!")
    }
</script>
```


CẤU TRÚC SWITCH-CASE



```
<script type="text/javascript">
var d = new Date()
theDay=d.getDay()
switch (theDay)
{
case 5:
    document.write("<b>Finally
Friday</b>")
    break
case 6:
    document.write("<b>Super
Saturday</b>")
    break
case 0:
    document.write("<b>Sleepy Sunday</b>")
    break
default:
    document.write("<b>I'm really looking
forward to this weekend!</b>")
}
</script>

<p>This JavaScript will generate a
different greeting based on what day it
is. Note that Sunday=0, Monday=1,
```

Finally Friday

This JavaScript will generate a different greeting based on what day it is. Note that Sunday=0, Monday=1, Tuesday=2, etc.

■ Cú pháp

```
for (initial-expression; condition; increment-expression) {
    statements
}
```

Ví dụ:

```
var n = 0;
for (var i = 0; i < 3; i++) {
    n += i;
    alert("The value of n is now " + n);
}
```

CẤU TRÚC LẶP FOR



```
<html>
<body>

<script type="text/javascript">
for (i = 0; i <= 5; i++)
{
document.write("The number is " + i)
document.write("<br>")
}
</script>

<p>Explanation:</p>

<p>This for loop starts with i=0.</p>

<p>As long as <b>i</b> is less than, or
equal to 5, the loop will continue to
run.</p>

<p><b>i</b> will increase by 1 each time
the loop runs.</p>

</body>
</html>
```

The number is 0
The number is 1
The number is 2
The number is 3
The number is 4
The number is 5

Explanation:

This for loop starts with i=0.

As long as **i** is less than, or equal to 5, the loop will continue to run.

i will increase by 1 each time the loop runs.

```
<html>
<body>

<script type="text/javascript">
for (i = 1; i <= 6; i++)
{
document.write("<h" + i + ">This is
header " + i)
document.write("</h" + i + ">")
}
</script>

</body>
</html>
```

This is header 1

This is header 2

This is header 3

This is header 4

This is header 5

This is header 6

CẤU TRÚC WHILE, DO-WHILE



■ Cú pháp

```
while (var<=endvalue)
```

```
{
```

```
    code to be executed
```

```
}
```

```
do
```

```
{
```

```
    code to be executed
```

```
}
```

```
while (var<=endvalue)
```

■ Ví dụ

```
<html>
<body>
  <script type="text/javascript">
    var i=0
    while (i<=10)
    {
      document.write("The number is " + i)
      document.write("<br />")
      i=i+1
    }
  </script>
</body>
</html>
```

■ Ví dụ

```
<html>
<body>
  <script type="text/javascript">
    var i=0
    do {
      document.write("The number is " + i)
      document.write("<br />")
      i=i+1
    }
    while (i<0)
  </script>
</body>
</html>
```

CẤU TRÚC WHILE, DO-WHILE



```
<body>

<script type="text/javascript">
i = 0
while (i <= 5)
{
document.write("The number is " + i)
document.write("<br>")
i++
}
</script>

<p>Explanation:</p>

<p><b>i</b> equal to 0.</p>

<p>While <b>i</b> is less than , or
equal to, 5, the loop will continue to
run.</p>

<p><b>i</b> will increase by 1 each
time the loop runs.</p>
```

The number is 0
The number is 1
The number is 2
The number is 3
The number is 4
The number is 5

Explanation:

i equal to 0.

While **i** is less than , or equal to, 5, the loop will continue to run.

i will increase by 1 each time the loop runs.

CẤU TRÚC WHILE, DO-WHILE



```
<script type="text/javascript">
i = 0
do
{
document.write("The number is " + i)
document.write("<br>")
i++
}
while (i <= 5)
</script>

<p>Explanation:</p>

<p><b>i</b> equal to 0.</p>

<p>The loop will run</p>

<p><b>i</b> will increase by 1 each time
the loop runs.</p>

<p>While <b>i</b> is less than , or
equal to, 5, the loop will continue to
run.</p>
```

The number is 0
The number is 1
The number is 2
The number is 3
The number is 4
The number is 5

Explanation:

i equal to 0.

The loop will run

i will increase by 1 each time the loop runs.

While **i** is less than , or equal to, 5, the loop will continue to run.

```
<html>
<body>
<script type="text/javascript">
var i=0
for (i=0;i<=10;i++)
{
if (i==3){break}
document.write("The number is " + i)
document.write("<br />")
}
</script>
</table>

<p>Explanation: The loop will break when
i=3.</p>

</body>
</html>
```

The number is 0
The number is 1
The number is 2

Explanation: The loop will break when i=3.



```
<html>
<body>
<script type="text/javascript">
var i=0
for (i=0;i<=10;i++)
{
if (i==3){continue}
document.write("The number is " + i)
document.write("<br />")
}
</script>
</table>
```

<p>Explanation: The loop will break the current loop and continue with the next value when i=3.</p>

```
</body>
</html>
```

The number is 0
The number is 1
The number is 2
The number is 4
The number is 5
The number is 6
The number is 7
The number is 8
The number is 9
The number is 10

Explanation: The loop will break the current loop and continue with the next value when i=3.

Giới thiệu về JavaScript

Biến, kiểu dữ liệu, phép toán

Lệnh điều khiển

Popup

Sử dụng các đối tượng

ALERT("SOMETEXT")



```
<html>
<head>
<script type="text/javascript">
function disp_alert()
{
alert("Hello again! This is how we" +
'\n' + "add line breaks to an alert
box!")
}
</script>
</head>

<body>
<form>
<input type="button"
onclick="disp_alert()" value="Display
alert box">
</form>
</body>

</html>
```

Display alert box

[JavaScript Application]



Hello again! This is how we
add line breaks to an alert box!

OK

CONFIRM("SOMETEXT")



```
<script type="text/javascript">
function disp_confirm()
{
var name=confirm("Press a button")
if (name==true)
{
document.write("You pressed the OK
button!")
}
else
{
document.write("You pressed the Cancel
button!")
}
}
</script>
</head>

<body>
<form>
<input type="button"
onclick="disp_confirm()" value="Display
a confirm box">
</form>
</body>
```

Display a confirm box

[JavaScript Application]



Press a button

OK

Cancel

```
<html>
<head>
<script type="text/javascript">
function disp_prompt()
{
var name=prompt("Please enter your
name","")
if (name!=null && name!="")
{
document.write("Hello " + name + "! How
are you today?")
}
}
</script>
</head>

<body>
<form>
<input type="button"
onclick="disp_prompt()" value="Display a
prompt box">
</form>
</body>

</html>
```

Display a prompt box

[JavaScript Application]



Please enter your name

OK

Cancel

KẾT HỢP VỚI FORM



```
<html>
<head>

<script type="text/javascript">
function myfunction()
{
alert("HELLO")
}
</script>

</head>
<body>

<form>
<input type="button"
onclick="myfunction()"
value="Call function">
</form>

<p>By pressing the button, a function
will be called. The function will alert
a message.</p>

</body>
</html>
```

Call function

By pressing the button, a function will be called. The function will alert a message.

[JavaScript Application]



HELLO

OK

KẾT HỢP VỚI FORM



```
<script type="text/javascript">
function myfunction(txt)
{
alert(txt)
}
</script>
</head>

<body>
<form>
<input type="button"
onclick="myfunction('Good Morning!')"
value="In the Morning">

<input type="button"
onclick="myfunction('Good Evening!')"
value="In the Evening">
</form>

<p>
When you click on one of the buttons, a
function will be called. The function
will alert
the argument that is passed to it.
</p>
```

In the Morning

In the Evening

When you click on one of the buttons, a function will be called. The function will alert the argument that is passed to it.

[JavaScript Application]



Good Evening!

OK

Giới thiệu về JavaScript

Biến, kiểu dữ liệu, phép toán

Lệnh điều khiển

Popup

Sử dụng các đối tượng

- Các đối tượng
 - String
 - Date
 - Array
 - Boolean
 - Math
 - HTML DOM

ĐỐI TƯỢNG STRING



String Object Methods

FF: Firefox, N: Netscape, IE: Internet Explorer

Method	Description	FF	N	IE
<u>anchor()</u>	Creates an HTML anchor	1	2	3
<u>big()</u>	Displays a string in a big font	1	2	3
<u>blink()</u>	Displays a blinking string	1	2	
<u>bold()</u>	Displays a string in bold	1	2	3
<u>charAt()</u>	Returns the character at a specified position	1	2	3
<u>charCodeAt()</u>	Returns the Unicode of the character at a specified position	1	4	4
<u>concat()</u>	Joins two or more strings	1	4	4
<u>fixed()</u>	Displays a string as teletype text	1	2	3
<u>fontcolor()</u>	Displays a string in a specified color	1	2	3
<u>fontsize()</u>	Displays a string in a specified size	1	2	3
<u>fromCharCode()</u>	Takes the specified Unicode values and returns a string	1	4	4
<u>indexOf()</u>	Returns the position of the first occurrence of a specified string value in a string	1	2	3
<u>italics()</u>	Displays a string in italic	1	2	3
<u>lastIndexOf()</u>	Returns the position of the last occurrence of a specified string value, searching backwards from the specified position in a string	1	2	3
<u>link()</u>	Displays a string as a hyperlink	1	2	3
<u>match()</u>	Searches for a specified string value in a string	1	4	4
<u>replace()</u>	Replaces some characters with some other characters in a string	1	4	4
<u>search()</u>	Searches a string for a specified value	1	4	4

ĐỐI TƯỢNG STRING



<u>slice()</u>	Extracts a part of a string and returns the extracted part in a new string	1	4	4
<u>small()</u>	Displays a string in a small font	1	2	3
<u>split()</u>	Splits a string into an array of strings	1	4	4
<u>strike()</u>	Displays a string with a strikethrough	1	2	3
<u>sub()</u>	Displays a string as subscript	1	2	3
<u>substr()</u>	Extracts a specified number of characters in a string, from a start index	1	4	4
<u>substring()</u>	Extracts the characters in a string between two specified indices	1	2	3
<u>sup()</u>	Displays a string as superscript	1	2	3
<u>toLowerCase()</u>	Displays a string in lowercase letters	1	2	3
<u>toUpperCase()</u>	Displays a string in uppercase letters	1	2	3
<u>toSource()</u>	Represents the source code of an object	1	4	-
<u>valueOf()</u>	Returns the primitive value of a String object	1	2	4

String Object Properties

Property	Description	FF	N	IE
<u>constructor</u>	A reference to the function that created the object	1	4	4
<u>length</u>	Returns the number of characters in a string	1	2	3
<u>prototype</u>	Allows you to add properties and methods to the object	1	2	4

■ Ví dụ

```
var txt="Hello world!"
document.write(txt.length)
document.write(txt.toUpperCase())
document.write(txt.substring(4, 8))
document.write(txt.small())
document.write(txt.strike())
document.write(txt.charAt(8))
```

ĐỐI TƯỢNG DATE



Date Object Methods

FF: Firefox, N: Netscape, IE: Internet Explorer

Method	Description	FF	N	IE
<u>Date()</u>	Returns today's date and time	1	2	3
<u>getDate()</u>	Returns the day of the month from a Date object (from 1-31)	1	2	3
<u>getDay()</u>	Returns the day of the week from a Date object (from 0-6)	1	2	3
<u>getMonth()</u>	Returns the month from a Date object (from 0-11)	1	2	3
<u>getFullYear()</u>	Returns the year, as a four-digit number, from a Date object	1	4	4
<u>getYear()</u>	Returns the year, as a two-digit or a four-digit number, from a Date object. Use <code>getFullYear()</code> instead !!	1	2	3
<u>getHours()</u>	Returns the hour of a Date object (from 0-23)	1	2	3
<u>getMinutes()</u>	Returns the minutes of a Date object (from 0-59)	1	2	3
<u>getSeconds()</u>	Returns the seconds of a Date object (from 0-59)	1	2	3
<u>getMilliseconds()</u>	Returns the milliseconds of a Date object (from 0-999)	1	4	4
<u>getTime()</u>	Returns the number of milliseconds since midnight Jan 1, 1970	1	2	3
<u>getTimezoneOffset()</u>	Returns the difference in minutes between local time and Greenwich Mean Time (GMT)	1	2	3
<u>getUTCDate()</u>	Returns the day of the month from a Date object according to universal time (from 1-31)	1	4	4

ĐỒI TƯỢNG DATE



Date Object Methods

FF: Firefox, N: Netscape, IE: Internet Explorer

Method	Description	FF	N	IE
<u>Date()</u>	Returns today's date and time	1	2	3
<u>getDate()</u>	Returns the day of the month from a Date object (from 1-31)	1	2	3
<u>getDay()</u>	Returns the day of the week from a Date object (from 0-6)	1	2	3
<u>getMonth()</u>	Returns the month from a Date object (from 0-11)	1	2	3
<u>getFullYear()</u>	Returns the year, as a four-digit number, from a Date object	1	4	4
<u>getYear()</u>	Returns the year, as a two-digit or a four-digit number, from a Date object. Use <u>getFullYear()</u> instead !!	1	2	3
<u>getHours()</u>	Returns the hour of a Date object (from 0-23)	1	2	3
<u>getMinutes()</u>	Returns the minutes of a Date object (from 0-59)	1	2	3
<u>getSeconds()</u>	Returns the seconds of a Date object (from 0-59)	1	2	3
<u>getMilliseconds()</u>	Returns the milliseconds of a Date object (from 0-999)	1	4	4
<u>getTime()</u>	Returns the number of milliseconds since midnight Jan 1, 1970	1	2	3
<u>getTimezoneOffset()</u>	Returns the difference in minutes between local time and Greenwich Mean Time (GMT)	1	2	3
<u>getUTCDate()</u>	Returns the day of the month from a Date object according to universal time (from 1-31)	1	4	4

ĐỐI TƯỢNG DATE



<u>getUTCDay()</u>	Returns the day of the week from a Date object according to universal time (from 0-6)	1	4	4
<u>getUTCMonth()</u>	Returns the month from a Date object according to universal time (from 0-11)	1	4	4
<u>getUTCFullYear()</u>	Returns the four-digit year from a Date object according to universal time	1	4	4
<u>getUTCHours()</u>	Returns the hour of a Date object according to universal time (from 0-23)	1	4	4
<u>getUTCMinutes()</u>	Returns the minutes of a Date object according to universal time (from 0-59)	1	4	4
<u>getUTCSeconds()</u>	Returns the seconds of a Date object according to universal time (from 0-59)	1	4	4
<u>getUTCMilliseconds()</u>	Returns the milliseconds of a Date object according to universal time (from 0-999)	1	4	4
<u>parse()</u>	Takes a date string and returns the number of milliseconds since midnight of January 1, 1970	1	2	3
<u>setDate()</u>	Sets the day of the month in a Date object (from 1-31)	1	2	3
<u>setMonth()</u>	Sets the month in a Date object (from 0-11)	1	2	3
<u>setFullYear()</u>	Sets the year in a Date object (four digits)	1	4	4
<u>setYear()</u>	Sets the year in the Date object (two or four digits). Use setFullYear() instead !!	1	2	3
<u>setHours()</u>	Sets the hour in a Date object (from 0-23)	1	2	3
<u>setMinutes()</u>	Set the minutes in a Date object (from 0-59)	1	2	3
<u>setSeconds()</u>	Sets the seconds in a Date object (from 0-59)	1	2	3
<u>setMilliseconds()</u>	Sets the milliseconds in a Date object (from 0-999)	1	4	4
<u>setTime()</u>	Calculates a date and time by adding or subtracting a specified number of milliseconds to/from midnight January 1, 1970	1	2	3

ĐỐI TƯỢNG DATE



<u>setUTCDate()</u>	Sets the day of the month in a Date object according to universal time (from 1-31)	1	4	4
<u>setUTCMonth()</u>	Sets the month in a Date object according to universal time (from 0-11)	1	4	4
<u>setUTCFullYear()</u>	Sets the year in a Date object according to universal time (four digits)	1	4	4
<u>setUTCHours()</u>	Sets the hour in a Date object according to universal time (from 0-23)	1	4	4
<u>setUTCMinutes()</u>	Set the minutes in a Date object according to universal time (from 0-59)	1	4	4
<u>setUTCSeconds()</u>	Set the seconds in a Date object according to universal time (from 0-59)	1	4	4
<u>setUTCMilliseconds()</u>	Sets the milliseconds in a Date object according to universal time (from 0-999)	1	4	4
<u>toSource()</u>	Represents the source code of an object	1	4	-
<u>toString()</u>	Converts a Date object to a string	1	2	4
<u>toGMTString()</u>	Converts a Date object, according to Greenwich time, to a string. Use toUTCString() instead !!	1	2	3
<u>toUTCString()</u>	Converts a Date object, according to universal time, to a string	1	4	4
<u>toLocaleString()</u>	Converts a Date object, according to local time, to a string	1	2	3
<u>UTC()</u>	Takes a date and returns the number of milliseconds since midnight of January 1, 1970 according to universal time	1	2	3
<u>valueOf()</u>	Returns the primitive value of a Date object	1	2	4

Date Object Properties

Property	Description	FF	N	IE
<u>constructor</u>	A reference to the function that created the object	1	4	4
<u>prototype</u>	Allows you to add properties and methods to the object	1	3	4

■ Ví dụ

```
var myDate=new Date()
myDate.setFullYear(2010,0,14)
myDate.setDate(myDate.getDate()+5)
myDate.setFullYear(2010,0,14)
var today = new Date()
if (myDate>today)
    alert("Today is before 14th January 2010")
else
    alert("Today is after 14th January 2010")
```

- ❖ Với lệnh khởi tạo Date không tham số, hệ thống sẽ trả về ngày hiện tại

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

- ❖ Khởi tạo đối tượng Date chỉ định rõ những tham số theo thứ tự sau: năm, tháng, ngày, giờ, phút, giây:

```
var vnowmanDay = new Date(2011, 9, 20);
```

```
var vnowmanDayAndTime = new Date(2011, 9, 20, 12, 20, 30);
```

- ❖ **Lưu ý:** trong Javascript, tháng được tính từ 0 - 11, nghĩa là 0 là tháng 1, 1 là tháng 2 ...

- ❖ Date còn được khởi tạo bằng cách truyền vào một chuỗi ngày tháng hợp lệ
 - `var date1 = new Date("10/20/2011");`
 - `var date2 = new Date("Wednesday, October 20, 2010");`
- ❖ Cả 2 kết quả đều như nhau, chỉ khác nội dung chuỗi tham số khởi tạo.
- ❖ Tháng trình bày trong chuỗi vẫn tính từ 1 - 12, không cần phải chuyển đổi thành 0 - 11 như truyền trực tiếp tham số tháng ở trên.
- ❖ Nhưng nếu ta sử dụng chuỗi làm tham số khởi tạo mà lại bỏ quên chữ `new`, kết quả sẽ không như mong đợi

CÁC PHƯƠNG THỨC GET



```
var now = new Date();
alert("Ngày hôm nay: " + now);
alert("Ngày hôm qua: " + new Date(now.getFullYear(), now.getMonth(),
now.getDate() - 1));
alert("Ngày mai: " + new Date(now.getFullYear(), now.getMonth(),
now.getDate() + 1));
alert("Ngày mốt: " + new Date(now.getFullYear(), now.getMonth(),
now.getDate() + 2));
alert("Tháng trước: " + new Date(now.getFullYear(), now.getMonth() -
1, now.getDate()).getMonth() + 1); // phải cộng thêm 1 vì tháng tính từ 0 - 11
alert("Tháng sau: " + new Date(now.getFullYear(), now.getMonth() + 1,
now.getDate()).getMonth() + 1); // phải cộng thêm 1 vì tháng tính từ 0 - 11
```

TÍNH KHOẢNG THỜI GIAN



```
var now = new Date();
var xmasDay = new Date(now.getFullYear(), 11, 24);
// khởi tạo ngày Noel cùng năm với ngày hiện tại
var offset = xmasDay.getTime() - now.getTime();
// lấy độ lệch của 2 mốc thời gian, đơn vị tính là millisecond
var totalDays = Math.round(offset / 1000 / 60 / 60 / 24);
alert("Còn " + totalDays + " ngày là đến Noel");
var totalHours = Math.round(offset / 1000 / 60 / 60);
alert("Còn " + totalHours + " giờ là đến Noel");
var totalMinutes = Math.round(offset / 1000 / 60);
alert("Còn " + totalMinutes + " phút là đến Noel");
var totalSeconds = Math.round(offset / 1000);
alert("Còn " + totalSeconds + " giây là đến Noel");
```

❖ //Tính chi tiết

```
var days = Math.floor(offset / 1000 / 60 / 60 / 24);
offset -= days * 1000 * 60 * 60 * 24; // giảm offset đi
var hours = Math.floor(offset / 1000 / 60 / 60);
offset -= hours * 1000 * 60 * 60; // giảm offset đi
var minutes = Math.floor(offset / 1000 / 60);
offset -= minutes * 1000 * 60;
var seconds = Math.floor(offset / 1000);
alert("Còn " + days + " ngày " + hours + " giờ " + minutes + " phút " +
seconds + " giây là đến Noel");
```

- ❖ Đoạn mã trên sử dụng hàm getTime của Date, hàm này trả về tổng số millisecond tính từ ngày 1/1/1970
- ❖ 2 ngày khác nhau nhưng cùng một mốc tính là 1/1/1970 chúng ta có thể tính toán ra được những thành phần khác

XÁC ĐỊNH NĂM NHUẦN



- ❖ Theo định nghĩa, năm nhuận là năm chia hết cho 100 và 400 ([xem thêm tại Wiki](#)), nhưng với Javascript ta chỉ cần:

```
function isLeapYear(year) {
    return new Date(year, 2, 0).getDate() > 28;
}
```

TÍNH SỐ NGÀY CỦA THÁNG



```
function getDaysOfMonth(year, month) {
    return new Date(year, month + 1, 0).getDate();
}
alert("Tháng 2 có " + getDaysOfMonth(new
Date().getFullYear(), 1) + " ngày"); // tháng 2 có giá trị là 1
alert("Tháng 10 có " + getDaysOfMonth(new
Date().getFullYear(), 9) + " ngày"); // tháng 10 có giá trị là 9
```

- ❖ Để xác định thứ tự một ngày trong năm, ý tưởng là sẽ lấy mốc thời gian hiện tại trừ đi mốc thời gian từ đầu năm

```
function getDayOfYear(date) {
    var offset = new Date(date.getFullYear(), date.getMonth(),
        date.getDate()).getTime() - new Date(date.getFullYear(), 0,
        1).getTime();
    return Math.ceil(offset / 1000 / 60 / 60 / 24);
}
```

- ❖ Tương tự chúng ta cũng có thể tính được thứ tự tuần trong năm bằng cách lấy thứ tự ngày chia và làm tròn cho 7

```
function getWeekOfYear(date) {
    var dayOfYear = getDayOfYear(date);
    return Math.ceil(dayOfYear / 7);
}
```

ĐỐI TƯỢNG ARRAY



Array Object Methods

FF: Firefox, N: Netscape, IE: Internet Explorer

Method	Description	FF	N	IE
<u>concat()</u>	Joins two or more arrays and returns the result	1	4	4
<u>join()</u>	Puts all the elements of an array into a string. The elements are separated by a specified delimiter	1	3	4
<u>pop()</u>	Removes and returns the last element of an array	1	4	5.5
<u>push()</u>	Adds one or more elements to the end of an array and returns the new length	1	4	5.5
<u>reverse()</u>	Reverses the order of the elements in an array	1	3	4
<u>shift()</u>	Removes and returns the first element of an array	1	4	5.5
<u>slice()</u>	Returns selected elements from an existing array	1	4	4
<u>sort()</u>	Sorts the elements of an array	1	3	4
<u>splice()</u>	Removes and adds new elements to an array	1	4	5.5
<u>toSource()</u>	Represents the source code of an object	1	4	-
<u>toString()</u>	Converts an array to a string and returns the result	1	3	4
<u>unshift()</u>	Adds one or more elements to the beginning of an array and returns the new length	1	4	-
<u>valueOf()</u>	Returns the primitive value of an Array object	1	2	4

Array Object Properties

Property	Description	FF	N	IE
<u>constructor</u>	A reference to the function that created the object	1	2	4
<u>index</u>		1	3	4
<u>input</u>		1	3	4
<u>length</u>	Sets or returns the number of elements in an array	1	2	4
<u>prototype</u>	Allows you to add properties and methods to the object	1	2	4

■ Ví dụ:

```
var mycars=new Array()
```

```
mycars[0]="Saab"
```

```
mycars[1]="Volvo"
```

```
mycars[2]="BMW"
```

```
var mycars=new Array("Saab","Volvo","BMW")
```

```
var mycars=new Array(3)
```

```
mycars[0]="Saab"
```

```
mycars[1]="Volvo"
```

```
mycars[2]="BMW"
```

■ Ví dụ

```
var arr = new Array(3)
arr[0] = "Jani"
arr[1] = "Tove"
arr[2] = "Hege"
var arr2 = new Array(3)
arr2[0] = "John"
arr2[1] = "Andy"
arr2[2] = "Wendy"
document.write(arr.concat(arr2))
```

■ Ví dụ

```
var arr = new Array(6)
arr[0] = "Jani"
arr[1] = "Hege"
arr[2] = "Stale"
arr[3] = "Kai Jim"
arr[4] = "Borge"
arr[5] = "Tove"
document.write(arr + "<br />")
document.write(arr.sort())
```


Boolean Object Methods

FF: Firefox, **N:** Netscape, **IE:** Internet Explorer

Method	Description	FF	N	IE
<u>toSource()</u>	Represents the source code of an object	1	4	-
<u>toString()</u>	Converts a Boolean value to a string and returns the result	1	4	4
<u>valueOf()</u>	Returns the primitive value of a Boolean object	1	4	4

Boolean Object Properties

Property	Description	FF	N	IE
<u>constructor</u>	A reference to the function that created the object	1	2	4
<u>prototype</u>	Allows you to add properties and methods to the object	1	2	4

■ Ví dụ

```
var myBoolean=new Boolean()
var myBoolean=new Boolean(0)
var myBoolean=new Boolean(null)
var myBoolean=new Boolean("")
var myBoolean=new Boolean(false)
var myBoolean=new Boolean(NaN)
var myBoolean=new Boolean(true)
var myBoolean=new Boolean("true")
var myBoolean=new Boolean("false")
var myBoolean=new Boolean("Richard")
```

ĐỐI TƯỢNG MATH



Math Object Methods

FF: Firefox, N: Netscape, IE: Internet Explorer

Method	Description	FF	N	IE
<u>abs(x)</u>	Returns the absolute value of a number	1	2	3
<u>acos(x)</u>	Returns the arccosine of a number	1	2	3
<u>asin(x)</u>	Returns the arcsine of a number	1	2	3
<u>atan(x)</u>	Returns the arctangent of x as a numeric value between -PI/2 and PI/2 radians	1	2	3
<u>atan2(y,x)</u>	Returns the angle theta of an (x,y) point as a numeric value between -PI and PI radians	1	2	3
<u>ceil(x)</u>	Returns the value of a number rounded upwards to the nearest integer	1	2	3
<u>cos(x)</u>	Returns the cosine of a number	1	2	3
<u>exp(x)</u>	Returns the value of E ^x	1	2	3
<u>floor(x)</u>	Returns the value of a number rounded downwards to the nearest integer	1	2	3
<u>log(x)</u>	Returns the natural logarithm (base E) of a number	1	2	3
<u>max(x,y)</u>	Returns the number with the highest value of x and y	1	2	3
<u>min(x,y)</u>	Returns the number with the lowest value of x and y	1	2	3
<u>pow(x,y)</u>	Returns the value of x to the power of y	1	2	3
<u>random()</u>	Returns a random number between 0 and 1	1	2	3
<u>round(x)</u>	Rounds a number to the nearest integer	1	2	3
<u>sin(x)</u>	Returns the sine of a number	1	2	3
<u>sqrt(x)</u>	Returns the square root of a number	1	2	3
<u>tan(x)</u>	Returns the tangent of an angle	1	2	3
<u>toSource()</u>	Represents the source code of an object	1	4	-
<u>valueOf()</u>	Returns the primitive value of a Math object	1	2	4

ĐỐI TƯỢNG MATH



Math Object Methods

FF: Firefox, N: Netscape, IE: Internet Explorer

Method	Description	FF	N	IE
<u>abs(x)</u>	Returns the absolute value of a number	1	2	3
<u>acos(x)</u>	Returns the arccosine of a number	1	2	3
<u>asin(x)</u>	Returns the arcsine of a number	1	2	3
<u>atan(x)</u>	Returns the arctangent of x as a numeric value between -PI/2 and PI/2 radians	1	2	3
<u>atan2(y,x)</u>	Returns the angle theta of an (x,y) point as a numeric value between -PI and PI radians	1	2	3
<u>ceil(x)</u>	Returns the value of a number rounded upwards to the nearest integer	1	2	3
<u>cos(x)</u>	Returns the cosine of a number	1	2	3
<u>exp(x)</u>	Returns the value of E ^x	1	2	3
<u>floor(x)</u>	Returns the value of a number rounded downwards to the nearest integer	1	2	3
<u>log(x)</u>	Returns the natural logarithm (base E) of a number	1	2	3
<u>max(x,y)</u>	Returns the number with the highest value of x and y	1	2	3
<u>min(x,y)</u>	Returns the number with the lowest value of x and y	1	2	3
<u>pow(x,y)</u>	Returns the value of x to the power of y	1	2	3
<u>random()</u>	Returns a random number between 0 and 1	1	2	3
<u>round(x)</u>	Rounds a number to the nearest integer	1	2	3
<u>sin(x)</u>	Returns the sine of a number	1	2	3
<u>sqrt(x)</u>	Returns the square root of a number	1	2	3
<u>tan(x)</u>	Returns the tangent of an angle	1	2	3
<u>toSource()</u>	Represents the source code of an object	1	4	-
<u>valueOf()</u>	Returns the primitive value of a Math object	1	2	4

Math Object Properties

Property	Description	FF	N	IE
<u>constructor</u>	A reference to the function that created the object	1	2	4
E	Returns Euler's constant (approx. 2.718)	1	2	3
LN2	Returns the natural logarithm of 2 (approx. 0.693)	1	2	3
LN10	Returns the natural logarithm of 10 (approx. 2.302)	1	2	3
LOG2E	Returns the base-2 logarithm of E (approx. 1.442)	1	2	3
LOG10E	Returns the base-10 logarithm of E (approx. 0.434)	1	2	3
PI	Returns PI (approx. 3.14159)	1	2	3
<u>prototype</u>	Allows you to add properties and methods to the object	1	2	4
SQRT1_2	Returns the square root of 1/2 (approx. 0.707)	1	2	3
SQRT2	Returns the square root of 2 (approx. 1.414)	1	2	3

■ Ví dụ

Math.E

Math.PI

Math.SQRT2

Math.SQRT1_2

Math.LN2

Math.LN10

Math.LOG2E

Math.LOG10E

document.write(Math.round(4.7))

document.write(Math.random())

document.write(Math.floor(Math.random()*11))

document.write(Math.round(-4.60))

ĐỐI TƯỢNG HTML DOM



Object	Description
<u>Anchor</u>	Represents an HTML a element (a hyperlink)
<u>Applet</u>	Represents an HTML applet element. The applet element is used to place executable content on a page
<u>Area</u>	Represents an area of an image-map. An image-map is an image with clickable regions
<u>Base</u>	Represents an HTML base element
<u>Basefont</u>	Represents an HTML basefont element
<u>Body</u>	Represents the body of the document (the HTML body)
<u>Button</u>	Represents a push button on an HTML form. For each instance of an HTML <input type="button"> tag on an HTML form, a Button object is created
<u>Checkbox</u>	Represents a checkbox on an HTML form. For each instance of an HTML <input type="checkbox"> tag on an HTML form, a Checkbox object is created
<u>Document</u>	Used to access all elements in a page
<u>Event</u>	Represents the state of an event, such as the element in which the event occurred, the state of the keyboard keys, the location of the mouse, and the state of the mouse buttons
<u>FileUpload</u>	For each instance of an HTML <input type="file"> tag on a form, a FileUpload object is created
<u>Form</u>	Forms are used to prompt users for input. Represents an HTML form element
<u>Frame</u>	Represents an HTML frame
<u>Frameset</u>	Represents an HTML frameset
<u>Hidden</u>	Represents a hidden field on an HTML form. For each instance of an HTML <input type="hidden"> tag on a form, a Hidden object is created
<u>History</u>	A predefined object which can be accessed through the history property of the Window object. This object consists of an array of URLs. These URLs are all the URLs the user has visited within a browser window

ĐỐI TƯỢNG HTML DOM



<u>Iframe</u>	Represents an HTML inline-frame
<u>Image</u>	Represents an HTML img element
<u>Link</u>	Represents an HTML link element. The link element can only be used within the <head> tag
<u>Location</u>	Contains information about the current URL
<u>Meta</u>	Represents an HTML meta element
<u>Navigator</u>	Contains information about the client browser
<u>Option</u>	Represents an option in a selection list on an HTML form. For each instance of an HTML <option> tag in a selection list on a form, an Option object is created
<u>Password</u>	Represents a password field on an HTML form. For each instance of an HTML <input type="password"> tag on a form, a Password object is created
<u>Radio</u>	Represents radio buttons on an HTML form. For each instance of an HTML <input type="radio"> tag on a form, a Radio object is created
<u>Reset</u>	Represents a reset button on an HTML form. For each instance of an HTML <input type="reset"> tag on a form, a Reset object is created
<u>Screen</u>	Automatically created by the JavaScript runtime engine and it contains information about the client's display screen
<u>Select</u>	Represents a selection list on an HTML form. For each instance of an HTML <select> tag on a form, a Select object is created

ĐỐI TƯỢNG HTML DOM



<u>Submit</u>	Represents a submit button on an HTML form. For each instance of an HTML <code><input type="submit"></code> tag on a form, a Submit object is created
<u>Table</u>	Represents an HTML table element
<u>TableData</u>	Represents an HTML td element
<u>TableHeader</u>	Represents an HTML th element
<u>TableRow</u>	Represents an HTML tr element
<u>Text</u>	Represents a text field on an HTML form. For each instance of an HTML <code><input type="text"></code> tag on a form, a Text object is created
<u>Textarea</u>	Represents an HTML textarea element
<u>Window</u>	Corresponds to the browser window. A Window object is created automatically with every instance of a <code><body></code> or <code><frameset></code> tag

```
<html>
<body>

<script type="text/javascript">

personObj=new Object()
personObj.firstname="John"
personObj.lastname="Doe"
personObj.age=50
personObj.eyecolor="blue"

document.write(personObj.firstname + " is " + personObj.age +
" years old.")

</script>

</body>
</html>
```

John is 50 years old.

```
<script type="text/javascript">
var c=0
var t
function timedCount()
{
document.getElementById('txt').value=c
c=c+1
t=setTimeout("timedCount()",1000)
}

function stopCount()
{
clearTimeout(t)
}
</script>
</head>

<body>
<form>
<input type="button" value="Start count!"
onClick="timedCount()">
<input type="text" id="txt">
<input type="button" value="Stop count!"
onClick="stopCount()">
</form>
```


Click on the "Start count!" button above to start the timer. The in
count for ever, starting at 0. Click on the "Stop count!" button to
counting.

```
<html>
<head>
<script type="text/javascript">
function startTime()
{
var today=new Date()
var h=today.getHours()
var m=today.getMinutes()
var s=today.getSeconds()
// add a zero in front of numbers<10
m=checkTime(m)
s=checkTime(s)
document.getElementById('txt').innerHTML=h+":"+m+":"+s
t=setTimeout('startTime()',500)
}

function checkTime(i)
{
if (i<10)
{
i="0" + i
}
return i
}
</script>
</head>
```

16:09:56

```
<script type="text/javascript">
var x = navigator
document.write("CodeName=" + x.appCodeName)
document.write("<br />")
document.write("MinorVersion=" + x.appMinorVersion)
document.write("<br />")
document.write("Name=" + x.appName)
document.write("<br />")
document.write("Version=" + x.appVersion)
document.write("<br />")
document.write("CookieEnabled=" + x.cookieEnabled)
document.write("<br />")
document.write("CPUClass=" + x.cpuClass)
document.write("<br />")
document.write("OnLine=" + x.onLine)
document.write("<br />")
document.write("Platform=" + x.platform)
document.write("<br />")
document.write("UA=" + x.userAgent)
document.write("<br />")
document.write("BrowserLanguage=" + x.browserLanguage)
document.write("<br />")
document.write("SystemLanguage=" + x.systemLanguage)
document.write("<br />")
document.write("UserLanguage=" + x.userLanguage)
```

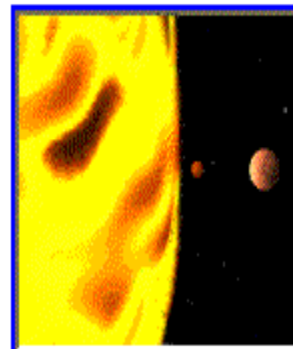
```
CodeName=Mozilla
MinorVersion=undefined
Name=Netscape
Version=5.0 (Windows; en-US)
CookieEnabled=true
CPUClass=undefined
OnLine=undefined
Platform=Win32
UA=Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.4)
Gecko/20030624 Netscape/7.1 (ax)
BrowserLanguage=undefined
SystemLanguage=undefined
UserLanguage=undefined
```

```
<script type="text/javascript">
function writeText(txt)
{
document.getElementById("desc").innerHTML=txt
}
</script>
</head>

<body>
<img src ="planets.gif" width ="145" height ="126"
alt="Planets" usemap="#planetmap" />

<map id ="planetmap" name="planetmap">
<area shape ="rect" coords ="0,0,82,126"
onMouseOver="writeText('The Sun and the gas giant planets
like Jupiter are by far the largest objects in our Solar
System.')"
href ="sun.htm" target ="_blank" alt="Sun" />

<area shape ="circle" coords ="90,58,3"
onMouseOver="writeText('The planet Mercury is very difficult
to study from the Earth because it is always so close to the
Sun.')"
href ="mercur.htm" target =" blank" alt="Mercury" />
```



The Sun and the gas giant planets like Jupiter are by far the Solar System.

- ❖ Biểu thức chính qui là một đối tượng mô tả mẫu các ký tự.
- ❖ Biểu thức chính qui được dùng để thực hiện so sánh sự trùng khớp mẫu và các chức năng “tìm và thay thế” trên văn bản.
- ❖ **Cú pháp**
 - **var txt=new RegExp(pattern,modifiers);** hay
 - **var txt=/pattern/modifiers;**
 - Pattern: xác định mẫu của biểu thức
 - Modifiers: xác định tìm kiếm nên là toàn cục, phân biệt hoa thường,...

THÍ DỤ 1: REGEXP



```
<html>
<body>
<script type="text/javascript">
var str = "Visit W3Schools";
var patt1 = /w3schools/i;
document.write(str.match(patt1));
</script>

</body>
</html>
```


THÍ DỤ 2: REGEXP



```
<html>
<body>
<script type="text/javascript">
var str="Is this all there is?";
var patt1=/is/g;
document.write(str.match(patt1));

</script>
</body>
</html>
```

THÍ DỤ 3: REGEXP



```
<html>
<body>
<script type="text/javascript">

var str="Is this all there is?";
var patt1=/is/gi;
document.write(str.match(patt1));

</script>
</body>
</html>
```

THÍ DỤ 4: REGEXP



```
<html>
<body>

<script type="text/javascript">
var patt1=new RegExp("e");

document.write(patt1.test("The best things in life are free"));
</script>

</body>
</html>
```

THÍ DỤ 5: REGEXP



```
<html>
<body>
<script type="text/javascript">
var patt1=new RegExp("e");

document.write(patt1.exec("The best things in life are
    free"));
</script>
</body></html>
```

Modifiers được dùng để thực hiện việc tìm kiếm không phân biệt hoa thường và toàn cục

Modifiers	Mô tả	FF	IE
i	Thực hiện so sánh không phân biệt hoa thường	1	4
g	Thực hiện so sánh toàn cục (tìm tất cả các trùng khớp hơn là dừng sau khi gặp trùng khớp đầu tiên)	1	4
m	Thực hiện so sánh nhiều dòng	1	4

BRACKET



Bracket ([]) được dùng để tìm dãy ký tự:

Biểu thức	Mô tả	F	IE
<code>[abc]</code>	Tìm bất cứ ký tự giữa hai dấu ngoặc	1	4
<code>[^abc]</code>	Tìm bất cứ ký tự không thuộc ký tự giữa hai dấu ngoặc	1	4
<code>[0-9]</code>	Tìm bất cứ con số nào từ 0 tới 9	1	4
<code>[a-z]</code>	Tìm bất cứ ký tự từ a tới z	1	4
<code>[A-Z]</code>	Tìm bất cứ ký tự từ A tới Z	1	4
<code>[a-Z]</code>	Tìm bất cứ ký tự từ a tới Z	1	4
<code>[adgk]</code>	Tìm bất cứ ký tự nào trong tập hợp được cho	1	4
<code>[^adgk]</code>	Tìm bất cứ ký tự nào ngoài tập hợp được cho	1	4
<code>[red blue green]</code>	Tìm một trong các khả năng được xác định: red, blue hay green	1	4

METACHARACTER



Metacharacters là những ký tự với ý nghĩa đặc biệt:

Metacharacter	Mô tả	F	IE
.	Tìm một ký tự, ngoại trừ dòng mới hay kết thúc dòng	1	4
\w	Tìm một ký tự là từ	1	4
\W	Tìm một ký tự không là từ	1	4
\d	Tìm một con số	1	4
\D	Tìm một ký tự không là con số	1	4
\s	Tìm một ký tự khoảng trắng	1	4
\S	Tìm một ký tự không là khoảng trắng	1	4
\b	Tìm sự trùng khớp tại vị trí bắt đầu/kết thúc của một từ	1	4
\B	Tìm sự trùng khớp tại vị trí KHÔNG bắt đầu/kết thúc của một từ	1	4
\0	Tìm một ký tự NUL	1	4

METACHARACTER



Metacharacter	Mô tả	F	IE
<u>\n</u>	Tìm một ký tự xuống dòng	1	4
<u>\f</u>	Tìm ký tự form feed	1	4
<u>\r</u>	Tìm ký tự trở về đầu dòng	1	4
<u>\t</u>	Tìm ký tự tab	1	4
<u>\v</u>	Tìm ký tự tab đứng	1	4
<u>\xxx</u>	Tìm ký tự được xác định bởi số bát phân xxx	1	4
<u>\xdd</u>	Tìm ký tự được xác định bởi một số thập lục phân dd	1	4
<u>\uxxxx</u>	Tìm ký tự Unicode được xác định bởi số thập lục phân xxxx	1	4

QUANTIFIER



Quantifier	Mô tả	F	IE
<u>n^+</u>	Tìm bất cứ chuỗi chứa ít nhất một n	1	4
<u>n^*</u>	Tìm bất cứ chuỗi chứa 0 hay nhiều lần n	1	4
<u>$n?$</u>	Tìm bất cứ chuỗi chứa 0 hay 1 lần n	1	4
<u>$n\{X\}$</u>	Tìm bất cứ chuỗi chứa một dãy X lần của n	1	4
<u>$n\{X,Y\}$</u>	Tìm bất cứ chuỗi chứa một dãy X hay Y lần của n	1	4
<u>$n\{X, \}$</u>	Tìm bất cứ chuỗi chứa một dãy ít nhất X lần của n	1	4
<u>$n\\$</u>	So sánh bất cứ chuỗi với n tại vị trí cuối của nó	1	4
<u>n</u>	So sánh bất cứ chuỗi với n tại vị trí đầu của nó	1	4
<u>$?=n$</u>	So sánh bất cứ chuỗi được theo sau bởi một chuỗi n	1	4
<u>$?!n$</u>	So sánh bất cứ chuỗi không được theo sau bởi một chuỗi n	1	4

Thuộc tính	Mô tả	F	IE
<u>global</u>	Xác định nếu "g" modifier được đặt	1	4
<u>ignoreCase</u>	Xác định nếu "i" modifier được đặt	1	4
<u>lastIndex</u>	Chỉ mục tại vị trí mà sự trùng khớp tiếp theo	1	4
<u>multiline</u>	Xác định nếu "m" modifier được đặt	1	4
<u>source</u>	Chuỗi của mẫu RegExp	1	4

Phương thức	Mô tả	F	IE
<u>compile()</u>	Thay đổi biểu thức chính quy	1	4
<u>exec()</u>	Tìm kiếm một chuỗi cho một giá trị xác định. Trả về giá trị được tìm thấy và ghi nhớ vị trí	1	4
<u>test()</u>	Tìm kiếm một chuỗi cho một giá trị xác định. Trả về true hay false	1	4

Hàm	Mô tả	F	IE
<u>decodeURI()</u>	Giải hóa một URI được mã hóa	1	5.5
<u>decodeURIComponent()</u>	Giải hóa một thành phần URI được mã hóa	1	5.5
<u>encodeURI()</u>	Mã hóa một chuỗi như một URI	1	5.5
<u>encodeURIComponent()</u>	Mã hóa một chuỗi như một thành phần URI	1	5.5
<u>escape()</u>	Mã hóa một chuỗi	1	3
<u>eval()</u>	Định giá trị một chuỗi và thực thi nó nếu như nó là một mã	1	3
<u>isFinite()</u>	Kiểm tra nếu giá trị là một số xác định	1	4
<u>isNaN()</u>	Kiểm tra nếu một giá trị không là một số	1	3
<u>Number()</u>	Chuyển giá trị của một đối tượng tới một số	1	
<u>parseFloat()</u>	Dịch một chuỗi và trả về số float	1	3
<u>parseInt()</u>	Dịch một chuỗi và trả về số integer	1	3
<u>String()</u>	Dịch giá trị của một đối tượng thành chuỗi	1	
<u>unescape()</u>	Giải mã một chuỗi được mã hóa bởi escape()	1	3

Thuộc tính	Mô tả	F	IE
<u>Infinity</u>	Một giá trị số biểu diễn số vô cùng dương hay âm	1	4
<u>NaN</u>	Hiển thị giá trị là “không là số” "Not a Number"	1	4
<u>undefined</u>	Hiển thị một biến không được gán giá trị	1	5.5

DHTML

JAVA
SCRIPT

LẬP TRÌNH
WEB

PHP

Cảm ơn!