

ways accessing elements from JS. txt

```
1. var  
fn=document.getElementById("fname").  
value;  
var len=fn.length;
```

```
2. var  
x=document.getElementById("form1");  
    var numofelements=x.length;  
    var key=x.elements[0].name;  
    var values=x.elements[0].value;  
    var len=values.length;
```

```
3. var  
fn=document.forms["form1"]["fname"].  
value;  
var len=fn.length;
```

```
4. var  
fn=document.form1.fname.value;  
//some browser is not supporting  
var len=fn.length;
```

ways accessing elements from JS.txt

```
5. var  
leftframedoc=parent.document.getElementById("leftframe").contentDocument  
;  
    var  
In=leftframedoc.getElementById("Iname").value;
```

Get Certified

Study Web Technologies
and get a diploma
at w3schools.com

JS & DOM Reference

Overview

JavaScript Objects

JS Array

JS Boolean

JS Date

JS Math

JS Number

JS String

JS RegExp

JS Global

Browser Objects

Window

Navigator

Screen

History

Location

Core DOM Objects

DOM Node

DOM NodeList

DOM NamedNodeMap

DOM Document

DOM Element

DOM Attr

HTML DOM Objects

DOM Document

DOM Events

DOM Elements

DOM Anchor

DOM Area

DOM Base

DOM Body

DOM Button

DOM Form

DOM Frame/IFrame

DOM Frameset

DOM Image

DOM Input Button

DOM Input Checkbox

DOM Input File

DOM Input Hidden

DOM Input Password

DOM Input Radio

DOM Input Reset

DOM Input Submit

DOM Input Text

DOM Link

DOM Meta

DOM Object

DOM Option

DOM Select

DOM Style

DOM Table

DOM td / th

DOM tr

DOM Textarea

JavaScript **RegExp** Object

« Previous

Next Reference »

RegExp Object

A regular expression is an object that describes a pattern of characters.

Regular expressions are used to perform pattern-matching and "search-and-replace" functions on text.

Syntax

```
var patt=new RegExp(pattern,modifiers);
```

or more simply:

```
var patt=/pattern/modifiers;
```

- pattern specifies the pattern of an expression
- modifiers specify if a search should be global, case-sensitive, etc.

For a tutorial about the RegExp object, read our [JavaScript RegExp Object tutorial](#).

Modifiers

Modifiers are used to perform case-insensitive and global searches:

Modifier	Description
i	Perform case-insensitive matching
g	Perform a global match (find all matches rather than stopping after the first match)
m	Perform multiline matching

Brackets

Brackets are used to find a range of characters:

Expression	Description
[abc]	Find any character between the brackets
[^abc]	Find any character not between the brackets
[0-9]	Find any digit from 0 to 9
[A-Z]	Find any character from uppercase A to uppercase Z
[a-z]	Find any character from lowercase a to lowercase z
[A-z]	Find any character from uppercase A to lowercase z
[adgk]	Find any character in the given set
[^adgk]	Find any character outside the given set
(red blue green)	Find any of the alternatives specified

Metacharacters

Metacharacters are characters with a special meaning:

Metacharacter	Description
.	Find a single character, except newline or line terminator
\w	Find a word character
\W	Find a non-word character
\d	Find a digit
\D	Find a non-digit character
\s	Find a whitespace character

SHARE THIS PAGE



Like 47k

WEB HOSTING

Best Web Hosting
eUK Web Hosting
UK Reseller Hosting
Cloud Hosting
Windows Server Host

WEB BUILDING

XML Editor - Free Trial!
FREE Website BUILDER
FREE Website Creator

STATISTICS

Browser Statistics
OS Statistics
Display Statistics

<u>\S</u>	Find a non-whitespace character
<u>\b</u>	Find a match at the beginning/end of a word
<u>\B</u>	Find a match not at the beginning/end of a word
<u>\0</u>	Find a NUL character
<u>\n</u>	Find a new line character
<u>\f</u>	Find a form feed character
<u>\r</u>	Find a carriage return character
<u>\t</u>	Find a tab character
<u>\v</u>	Find a vertical tab character
<u>\xxx</u>	Find the character specified by an octal number xxx
<u>\xdd</u>	Find the character specified by a hexadecimal number dd
<u>\uxxxx</u>	Find the Unicode character specified by a hexadecimal number xxxx

Quantifiers

Quantifier	Description
<u>n+</u>	Matches any string that contains at least one n
<u>n*</u>	Matches any string that contains zero or more occurrences of n
<u>n?</u>	Matches any string that contains zero or one occurrences of n
<u>n{X}</u>	Matches any string that contains a sequence of X n's
<u>n{X,Y}</u>	Matches any string that contains a sequence of X to Y n's
<u>n{X,}</u>	Matches any string that contains a sequence of at least X n's
<u>n\$</u>	Matches any string with n at the end of it
<u>^n</u>	Matches any string with n at the beginning of it
<u>?=n</u>	Matches any string that is followed by a specific string n
<u>?!n</u>	Matches any string that is not followed by a specific string n

RegExp Object Properties

Property	Description
<u>global</u>	Specifies if the "g" modifier is set
<u>ignoreCase</u>	Specifies if the "i" modifier is set
<u>lastIndex</u>	The index at which to start the next match
<u>multiline</u>	Specifies if the "m" modifier is set
<u>source</u>	The text of the RegExp pattern

RegExp Object Methods

Method	Description
<u>compile()</u>	Compiles a regular expression
<u>exec()</u>	Tests for a match in a string. Returns the first match
<u>test()</u>	Tests for a match in a string. Returns true or false

[<< Previous](#)

[Next Reference >>](#)

22,756,521 Sites Built with Wix. Create yours today!

[Click here](#) to make a stunning website for Free.

Wix.com offers an easy-to-use online platform that enables anyone to design and publish professional, beautiful websites for free.

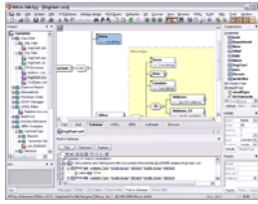
With total design control, eCommerce features and superior search engine results, Wix offers a complete solution for launching an exquisite website while saving tons of money.

Altova® XMLSpy® - The industry's leading XML editor!



Whether you're new to XML or already an advanced user, the user-friendly views and powerful entry helpers, wizards, and debuggers in XMLSpy are designed to meet your XML and Web development needs from start to finish. **New features in Version 2013!**

- XML editor
- Graphical XML Schema / DTD editors
- Smart Fix XML validation with automatic error correction
- XSLT 1.0/2.0 editor, debugger, profiler
- XQuery editor, debugger, profiler
- XBRL validator, taxonomy editor, taxonomy wizard
- New! Chart creation for XML data
- Support for Office Open XML (OOXML)
- Graphical WSDL 1.1/2.0 editor & SOAP debugger
- JSON editing & conversion
- Java, C#, C++ code generation
- 32-bit and 64-bit versions
- HTML5 and CSS3 support



Try it now!

Download a free trial today!



Top 10 Tutorials

- » [HTML Tutorial](#)
- » [HTML5 Tutorial](#)
- » [CSS Tutorial](#)
- » [CSS3 Tutorial](#)
- » [JavaScript Tutorial](#)
- » [jQuery Tutorial](#)
- » [SQL Tutorial](#)
- » [PHP Tutorial](#)
- » [ASP.NET Tutorial](#)
- » [XML Tutorial](#)

Top 10 References

- » [HTML/HTML5 Reference](#)
- » [CSS 1,2,3 Reference](#)
- » [CSS 3 Browser Support](#)
- » [JavaScript](#)
- » [HTML DOM](#)
- » [XML DOM](#)
- » [PHP Reference](#)
- » [jQuery Reference](#)
- » [ASP.NET Reference](#)
- » [HTML Colors](#)

Examples

- » [HTML Examples](#)
- » [CSS Examples](#)
- » [XML Examples](#)
- » [JavaScript Examples](#)
- » [HTML DOM Examples](#)
- » [XML DOM Examples](#)
- » [AJAX Examples](#)
- » [ASP.NET Examples](#)
- » [Razor Examples](#)
- » [ASP Examples](#)
- » [SVG Examples](#)

Quizzes

- » [HTML Quiz](#)
- » [HTML5 Quiz](#)
- » [XHTML Quiz](#)
- » [CSS Quiz](#)
- » [JavaScript Quiz](#)
- » [jQuery Quiz](#)
- » [XML Quiz](#)
- » [ASP Quiz](#)
- » [PHP Quiz](#)
- » [SQL Quiz](#)

Color Picker



Statistics

- » [Browser Statistics](#)
- » [Browser OS](#)
- » [Browser Display](#)



[REPORT ERROR](#) | [HOME](#) | [TOP](#) | [PRINT](#) | [FORUM](#) | [ABOUT](#) | [ADVERTISE WITH US](#)

W3Schools is optimized for learning, testing, and training. Examples might be simplified to improve reading and basic understanding. Tutorials, references, and examples are constantly reviewed to avoid errors, but we cannot warrant full correctness of all content. While using this site, you agree to have read and accepted our [terms of use](#) and [privacy policy](#). Copyright 1999-2013 by Refsnes Data. All Rights Reserved.

Regular Expression Examples

operator	description	sample pattern	matches	doesn't match
.	any character but newline	.	e	\n
^	beginning of string	^a	apple	banana
\$	end of string	a\$	banana	apple
[characters]	any characters in braces	[abcABC]	a	d
[char range]	describe range of characters	[a-zA-z]	r	9
\d	any digit	\d\d\d-\d\d\d\d	123-4567	the-thing
\b	word boundary	\bthe\b	the	theater
+	one or more occurrences of preceding character	\d+	1234	text
*	zero or more occurrences of preceding character	[a-zA-z]\d*	f16, b	9
{digit}	repeat preceding character that many times	\d{3}-\d{4}	123-4567	999-99-9999
	or operator	apple banana	apple, banana	peach
(pattern segment)	store results in pattern memory returned with numeric code	(^.).*/1	gig, blab (any word that starts and ends w/ same letter)	any other word