

Regular Expressions

Symbol	Correspondence	Example
\	Escape character	<code>[\.]</code> contain an "."
^	Start the line	<code>^b\$</code> contain only b
.	Any character	<code>^.\$</code> contain one character .
\$	End of line	<code>er\$</code> ended by "er"
	Alternative	<code>^(a A)</code> start by a or A
()	Groups	<code>^((a) (er))</code> start by a or er
-	Range of characters	<code>^[a-d]</code> start by a,b,c or d
[]	a set of characters	<code>[0-9]</code> contents a number
[^]	All except a set of characters	<code>^[^a]</code> is not starting with a
+	1 time or more	<code>^(a)+</code> start with one or more a
?	0 or 1 times	<code>^(a)?</code> start or not with a
*	0 or more times	<code>^(a)*</code> can or not start with a
{x}	x times exact	<code>a{2}</code> 2 times "a"
{x,}	x times at least	<code>a{2,}</code> at least 2 times "a"
{x, y}	minimum x times, maximum y	<code>a{2,4}</code> 2, 3 or 4 times "a"

Alias	Correspondence	Equivalence
\n	Newline character	
\r	Newline character	
\t	Tab character	
\s	character space (space, tab , etc)	<code>[\f\n\r\t\v]</code>
\S	All but one space	<code>[^\f\n\r\t\v]</code>
\d	A number	<code>[0-9]</code>
\D	All but a number	<code>[^0-9]</code>
\w	One character	<code>[a-zA-Z0-9_]</code>
\W	All but a character	<code>[^a-zA-Z0-9_]</code>

Examples

A string that contains letters from a to d or any uppercase letters: `\[a-zA-Z]\`

String starting (and ending) by y or z or Y or Z: `\^(y | z)$\i`

String containing no figure: `\[^0-9]\`

String containing the digits 1 or 2 or symbol ^ : `\[12\^\]`

You can use several metacharacters: `\^[pP]hara(onix)?$ \g`

String containing: phara, Phara, pharaonix or Pharaonix

JS - Regular expression - Boolean method: `.test()`

Use this web link to test the Regex : <http://regexr.com/>

Cookie

You can use those function to create (set) and to read (get) cookies. Use any other browser but Chrome to test set and get of your cookies.

```
function setCookie(cname, cvalue, nbrSec) {
    var d = new Date();
    d.setTime(d.getTime() + (nbrSec*1000));
    var expires = "expires="+d.toUTCString();
    document.cookie = cname + "=" + cvalue + ";" + expires;
}

function getCookie(cname) {
    var name = cname + "=";
    var ca = document.cookie.split(';');
    for(var i=0; i<ca.length; i++) {
        var c = ca[i];
        while (c.charAt(0)==' ') c = c.substring(1);
        if (c.indexOf(name) == 0) return c.substring(name.length, c.length);
    }
    return "";
}
```

Examples

```
<input type="button" onclick="setCookie('cookie_name',c_value,3600);" value="Create
cookie for one hour" />
<input type="button" onclick="alert(getCookie('cookie_name'));" value="Show cookie value"
/>
```

jQuery Selectors

Use our [jQuery Selector Tester](#) to demonstrate the different selectors.

Selector	Example	Selects
<code>*</code>	<code>\$("*")</code>	All elements
<code>#id</code>	<code>\$("#lastname")</code>	The element with id="lastname"
<code>.class</code>	<code>\$(".intro")</code>	All elements with class="intro"
<code>.class, .class</code>	<code>\$(".intro, .demo")</code>	All elements with the class "intro" or "demo"
<code>element</code>	<code>\$("p")</code>	All <p> elements
<code>el1, el2, el3</code>	<code>\$("h1, div, p")</code>	All <h1>, <div> and <p> elements
<code>:first</code>	<code>\$("p:first")</code>	The first <p> element
<code>:last</code>	<code>\$("p:last")</code>	The last <p> element
<code>:even</code>	<code>\$("tr:even")</code>	All even <tr> elements
<code>:odd</code>	<code>\$("tr:odd")</code>	All odd <tr> elements
<code>:focus</code>	<code>\$(":focus")</code>	The element that currently has focus
<code>:contains(text)</code>	<code>\$(":contains('Hello'))"</code>	All elements which contains the text "Hello"
<code>:empty</code>	<code>\$(":empty")</code>	All elements that are empty
<code>:parent</code>	<code>\$(":parent")</code>	All elements that are a parent of another element
<code>:hidden</code>	<code>\$("p:hidden")</code>	All hidden <p> elements
<code>:visible</code>	<code>\$("table:visible")</code>	All visible tables
<code>[attribute]</code>	<code>\$("[href]")</code>	All elements with a href attribute
<code>[attribute=value]</code>	<code>\$("[href='default.htm']")</code>	All elements with a href attribute value equal to "default.htm"
<code>[attribute!=value]</code>	<code>\$("[href!='default.htm']")</code>	All elements with a href attribute value not equal to "default.htm"
<code>[attribute\$=value]</code>	<code>\$("[href\$='.jpg']")</code>	All elements with a href attribute value ending with ".jpg"
<code>:input</code>	<code>\$(":input")</code>	All input elements
<code>:text</code>	<code>\$(":text")</code>	All input elements with type="text"

:password	\$(":password")	All input elements with type="password"
:radio	\$(":radio")	All input elements with type="radio"
:checkbox	\$(":checkbox")	All input elements with type="checkbox"
:submit	\$(":submit")	All input elements with type="submit"
:button	\$(":button")	All input elements with type="button"
:image	\$(":image")	All input elements with type="image"
:file	\$(":file")	All input elements with type="file"
:enabled	\$(":enabled")	All enabled input elements
:disabled	\$(":disabled")	All disabled input elements
:selected	\$(":selected")	All selected input elements
:checked	\$(":checked")	All checked input elements

Example

```
<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
<script>
$(document).ready(function(){
  $(":input").css("background-color", "red");
});
</script>
</head>
<body>

<form action="">
  Name: <input type="text" name="user"><br>
  Password: <input type="password" name="password"><br>
  <button type="button">Useless Button</button>
  <input type="button" value="Another useless button"><br>
  <input type="reset" value="Reset">
  <input type="submit" value="Submit"><br>
</form>

</body>
</html>
```