Javascript Regular Expressions

- description : Javascript Regular Expressions
- author :
- email : shlim@repia.com
- lastupdate : 2021-04-22

The source of this article

[Javascript Regular Expressions](#)

(Regular Expression,)   .

replace   .

What is a Regular Expression?

(Regular Expression,)   .

,   .

.

(text search)   **(text replace)**

.

Syntax

```
/pattern/modifiers;
```

Example

```
var patt = /w3schools/i;
```

:

/w3schools/i   (   )   .

w3schools ( ) .

i ( ) (modifier) .

Using String Methods

JavaScript , search() replace() .

search() , .

replace() .

Using String search() With a String

search() .

Example

["W3schools" string .

```html
<!DOCTYPE html>
<html>
<body>
  <h2>JavaScript String Methods</h2>
  <p>Search a string for "W3schools", and display the position of the
match.</p>
  <p id="demo"></p>   <!-- 6 -->
  <script>
    let str = "Visit W3Schools!";
    let n = str.search("W3Schools");  /* W3Schools                  ? */
    document.getElementById("demo").innerHTML = n;
  </script>
</body>
</html>
```

Using String search() With a Regular Expression

Example

"w3schools" .

```html
<!DOCTYPE html>
```

```
<html>
<body>
  <h2>JavaScript Regular Expressions</h2>
  <p>Search a string for "w3schools", and display the position of the
match:</p>
  <p id="demo"></p>    <!-- 6 -->
  <script>
    let str = "Visit W3Schools!";
    let n = str.search(/w3Schools/i);  /*
  ? */
    document.getElementById("demo").innerHTML = n;
  </script>
</body>
</html>
```

Using String replace() With a String

replace()   .

```
<!DOCTYPE html>
<html>
<body>
  <h2>JavaScript String Methods</h2>
  <p>Replace "Microsoft" with "W3Schools" in the paragraph below:</p>
  <button onclick="myFunction()">Try it</button>
  <p id="demo">Please visit Microsoft!</p>    <!-- Please visit W3Schools! -
->
  <script>
    function myFunction() {
      let str = document.getElementById("demo").innerHTML;
      let txt = str.replace("Microsoft", "W3Schools");
      document.getElementById("demo").innerHTML = txt;
    }
  </script>
</body>
</html>
```

Use String replace() With a Regular Expression

(a case insensitive regular expression)   Microsoft
      W3Schools   .

```
<!DOCTYPE html>
<html>
<body>
```

```
<h2>JavaScript Regular Expressions</h2>
<p>Replace "Microsoft" with "W3Schools" in the paragraph below:</p>
<button onclick="myFunction()">Try it</button>
<p id="demo">Please visit Microsoft and Microsoft!</p>
<script>
  function myFunction() {
    let str = document.getElementById("demo").innerHTML;
    let txt = str.replace(/microsoft/i, "W3Schools");
    document.getElementById("demo").innerHTML = txt;
  }
</script>
</body>
</html>
```

Did You Notice?

,   (   )   .
                              (:   ).

Regular Expression Modifiers

(Modifiers)   .

Modifier	Description
i	.
g	( )
m	.

Example of i modifier

```
<script>
    function myFunction() {
      let str = "Visit W3Schools";
      let patt1 = /w3schools/i; /*              w3schools        patt1
*/
      let result = str.match(patt1); /*    str         patt1(
w3schools                    result     ) */
      document.getElementById("demo").innerHTML = result; /*      result
id="demo"        p            */
    }
  </script>
```

Example of g modifier

```html
<script>
    function myFunction() {
      let str = "Is this all there is?";
      let patt1 = /is/g;   /*                          */
      let result = str.match(patt1);
      document.getElementById("demo").innerHTML = result;
    }
```

Example of m modifier

```html
  <script>
    function myFunction() {
      let str = "\nIs th\nis it?";
      let patt1 = /^is/m;
      let result = str.match(patt1);
      document.getElementById("demo").innerHTML = result;  /* is */
    }
  </script>
```

Regular Expression Patterns

:

Expression	Description
[abc]	.
[0-9]	.
(x|y)	| alternatives .

Example of [abc] pattern

```html
<script>
    function myFunction() {
      let str = "Is this all there is?";
      let patt1 = /[h]/g;
      let result = str.match(patt1);
      document.getElementById("demo").innerHTML = result; /* h,h */
    }
</script>
```

Example of [0-9] pattern

```html
  <script>
```

```
    function myFunction() {
      let str = "123456789";
      let patt1 = /[1-4]/g; /*    str    1~4              patt1      */
      var result = str.match(patt1);  /*    patt1                str
        result      */
      document.getElementById("demo").innerHTML = result; /* 1,2,3,4 */
    }
  </script>
```

Example of (x|y) pattern

```
  <script>
    function myFunction() {
      let str = "re, green, red, green, gren, gr, blue, yellow";
      let patt1 = /(red|green)/g;
      let result = str.match(patt1);
      document.getElementById("demo").innerHTML = result; /* green,red,green
*/
    }
  </script>
```

Metacharacters are characters with a special meaning:

Metacharacter(   ) :

Metacharacter	Description
\d	(digit) . digit: 0 9 10
\s	(whitespace character)
\b	\ bWORD WORD \ b
\uxxx	16 xxxx .

Example of Metachatacters (\d)

Example of Metacharacters (\s)

Exmaple of Metacharacters (\b)

Example of Metacharacters (\uxxxx)

Quantifiers define quantities

Quantifier	Description
n+	n n .
n*	n O .
n?	n O 1 .

Example of Quantifiers (n+)

Example of Quantifiers (n*)

Example of Quantifiers (n?)

Using the RegExp Object

, [Javascript](#), [Regular](#), [Expressions](#)