

Javascript. Modul de lucru cu clasa String

Obiectele clasei String sunt folosite pentru a manipula secvențe de caractere (text). Un obiect String este creat cu instrucțiunea `new String()`.

Sintaxa:

```
var txt = new String("sir de caractere");
```

sau mai simplu:

```
var txt = "sir de caractere";
```

Metodele clasei String

length - returneaza numarul de caractere dintr-un sir (string)

anchor() - Returneaza un sir ca si "anchor"

big() - Returneaza un sir cu text mare

blink() - Returneaza un sir care clipeste

bold() - Returneaza un sir cu litere ingrosate

charAt() - Returneaza un caracter de la pozitia care este specificata

charCodeAt() - Returneaza codul ASCII al unui caracter de la o pozitie specificata

concat() - Returneaza doua siruri concatenate

fixed() - Returneaza un sir cu caractere tip

fontcolor() - Returneaza un sir cu o culoare specificata

fontsize() - Returneaza un sir cu litere de o anume marime

fromCharCode() - Returneaza valoare Unicode a unui caracter

indexOf() - Returneaza pozitia primei aparitii a unui subsir in un sir, sau daca subsirul nu e gasit, valoarea -1

italics() - Returneaza un sir in italic (scris aplecat)

lastIndexOf() - Returneaza pozitia primei aparitii a unui subsir in un sir, sau daca acesta nu e gasit, -1 (Important: se incepe din dreapta sirului spre stanga)

link() - Returneaza un sir ca hyperlink

match() - Similar cu `indexOf` si `lastIndexOf`, dar aceasta metoda returneaza sirul specificat sir, sau "null", in locul unor valori numerice

replace() - Inlocuieste intr-un sir unele caractere specificate cu altele noi specificate.

search() - Returneaza un numar intreg daca sirul contine caracterele specificate, altfel returneaza -1

slice() - Returneaza un sir incepand de la pozitia index specificata

small() - Returneaza un sir cu caractere mai mici

split() - Imparte un sir in mai multe siruri, in functie de caracterele specificate

strike() - Returneaza un sir taiat cu o linie la mijloc

sub() - Returneaza un sir ca indice

substr() - Returneaza un subsir specificat astfel, exemplu: 12,8 returneaza 8 caractere, incepand de la caracterul al 12-lea (se incepe de la 0)

substring() - Returneaza un subsir specificat astfel, exemplu: 8,12 returneaza toate caracterele, incepand de la caracterul al 8-lea, pana la al 12-lea

sup() - Returneaza un sir ca putere (superscript)

toFixed(n) - Returneaza sirul numeric rotunjindu-l la o valoare cu 'n' zecimale

toLowerCase() - Converteste un sir in litere mici

toUpperCase() - Converteste un sir in litere mari

Exemplul 1

```
<html>

<body>

<h3>Obiectul String. Determinarea lungimii unui sir</h3>

<Br/>

<script type="text/javascript">

var txt="Bine ati venit!";

document.write("Sirul este: "+txt+"<br/>");

document.write("Are lungimea "+txt.length);

</script>

<p>

<b>Obs.</b>

Sirul nu se modifica.

</p>

</body>

</html>
```

Exemplul 2

```
<html>

<body>

<h3>Obiectul String. Utilizarea tagurilor HTML pentru stilizarea unui sir.</h3>

<Br/>

<script type="text/javascript">

var txt="Bine ati venit!";

document.write("<p>Big: " + txt.big() + "</p>"); document.write("<p>Small: " + txt.small() + "</p>");

document.write("<p>Bold: " + txt.bold() + "</p>"); document.write("<p>Italic: " + txt.italics() + "</p>");
```

```

document.write("<p>Blink: " + txt.blink() + " (nu functioneaza in IE, Chrome, Safari)</p>");
document.write("<p>Fixed: " + txt.fixed() + "</p>"); document.write("<p>Strike: " + txt.strike() + "</p>");

document.write("<p>Fontcolor: " + txt.fontcolor("Blue") + "</p>"); document.write("<p>Fontsize: " +
txt.fontSize(14) + "</p>");

document.write("<p>Subscript: " + txt.sub() + "</p>"); document.write("<p>Superscript: " + txt.sup() +
"</p>");

document.write("<p>Link: " + txt.link("http://www.google.com") + "</p>"); </script> <br/> <br/>
<p><b>Obs.</b>Sirul stilizat nu se modifica!</p>

</body>

</html>

```

Exemplul 3

Ilustrează cum se utilizează metoda indexOf() pentru a determina poziția primei apariții a unei valori într-un șir.

```

<html> <body>

<h3>Obiectul String. Cautarea primei aparitii a unei valori in sir cu indexof().</h3>

<Br/>

<script type="text/javascript">

var str="Buna ziua!";

document.write("Sirul in care se cauta este: "+str+"<br/>");

document.write("Sirul \"Buna\" apare in sir in pozitia "+str.indexOf("Buna") + "<br />");
document.write("Sirul \"ZIUA\" apare in sir in pozitia "+str.indexOf("ZIUA") + "<br />");
document.write("Sirul \"ziua\" apare in sir in pozitia "+str.indexOf("ziua"));

</script>

<p>

<b>Obs.</b>

Sirul nu se modifica in urma cautarii!

</p>

```

</body>

</html>

Javascript. Modul de lucru cu obiecte Array

Crearea unui tablou

Un tablou poate fi definit în trei moduri:

1:

```
var pets=new Array(); //tablou obisnuit
```

```
pets[0]="Caine";
```

```
pets[1]="Pisica";
```

```
pets[2]="Papagal";
```

2:

```
var pets=new Array("Caine","Pisica","Papagal"); //tablou condensat
```

3:

```
var pets=["Caine","Pisica","Papagal"]; //tablou literal
```

Obs: Dacă în tablou stocați valori numerice sau logice, tipul tabloului va fi Number sau Boolean, în loc de String.

Accesarea elementelor dintr-un tablou

Puteți accesa un element dintr-un tablou precizând numele tabloului și indicele elementului. Primul element din tablou are indicele 0. Următoarea linie de cod

```
document.write(pets[0]);
```

va afișa șirul: Caine

Pentru a afla numărul de elemente ale unui **Array**, se folosește proprietatea "**length**", ca în exemplu următor:

- `nr_colegi = nume_colegi.length`

Obiectul **Array** are următoarele metode:

concat() - Returneaza un tablou rezultat din concatenarea a doua tablouri
join() - Returneaza un tablou format din toate elementele unui tablou concatenat
reverse() - Returneaza inversul unui tablou
slice() - Returneaza o parte specificata a unui tablou
sort() - Returneaza tabloul ordonat

Exemplul 1

Urmatorul exemplu foloseste o instructiune "for" care parcurge un tablou (aici tabloul "nume_colegi"), extrage valoarea fiecarui element si o afiseaza.

```
<script type="text/javascript">
<!--
var nume_colegi = new Array();
nume_colegi[0] = "Cristi";
nume_colegi[1] = "Ion";
nume_colegi[2] = "Simona";
nume_colegi[3] = "Adi";

var nr_elemente = nume_colegi.length;      // Afla nr. elemente din matrice

document.write("Afiseaza numele fiecarui coleg: <br />");
for (i=0; i<nr_elemente; i++) {
    document.write(nume_colegi[i] + "<br />");
}
//-->
</script>
```

Exemplul 2

Folosim acelasi exemplu de mai sus.

"sort()" sorteaza elementele din Array in ordine alfabetica (sau crescatoare in cazul valorilor numerice).

```
<script type="text/javascript">
<!--
var nume_colegi = new Array();
nume_colegi[0] = "Cristi";
nume_colegi[1] = "Ion";
nume_colegi[2] = "Simona";
nume_colegi[3] = "Adi";

var nr_elemente = nume_colegi.length;      // Afla nr. elemente din matrice
nume_colegi.sort();                        // Aranjeaza elementele in ordine crescatoare, alfabetic

document.write("Afiseaza numele in ordine alfabetica: <br />");
```

```
for (i=0; i<nr_elemente; i++) {  
    document.write(ume_colegi[i] + "<br />");  
}  
//-->  
</script>
```

Tema

1. Verificati exemplele anterioare.
2. Memorati numele si adresa de email pentru 5 persoane in obiecte String. Apoi afisati intr-un table cu trei coloane aceste informatii (nr.crt/nume/email).
3. Pentru un sir de caractere care contine doua cuvinte separate printr-un spatiu memorat intr-un obiect String afisati cuvintele pe randuri diferite.
4. Pentru un text dat stergeti din el spatiile inutile dintre cuvinte.
5. Creati un vector cu primele n numere impare. Apoi afisati numerele prime in ordine crescatoare, respective descrecatoare parcurgand acest vector.
6. Creati un vector cu primele 30 de elemente din sirul lui Fibonacci. Apoi afisati componentele pare pe o linie si cele impare pe o alta linie.