



Интернет програмирање предавање 08

Проф. др Мирослав Лутовац
mlutovac@viser.edu.rs



JavaScript Date metode

- Date metode omogućavaju da se preuzme i postavi vrednost za datum i vreme (godina, mesec, dan, sat, minut, sekunda, ms)
- getDate() preuzima dan u mesecu kao broj (1-31)
- getDay() preuzima dan u nedelji kao broj (0-6)
- getFullYear() preuzima godinu sa 4 cifre (yyyy)
- getHours() preuzima sat (0-23)
- getMilliseconds() preuzima milisekunde (0-999)
- getMinutes() preuzima minute (0-59)
- getMonth() preuzima mesec (0-11)
- getSeconds() preuzima sekunde (0-59)
- getTime() preuzima vreme u ms od 1.1.1970.



Date metode da se preuzme

Interni klok JavaScript počinje od ponoći 1.1.1970.

- getDate()** preuzima dan u mesecu kao broj (1-31)
- getDay()** preuzima dan u nedelji kao broj (0-6)
- getFullYear()** preuzima godinu sa 4 cifre (yyyy)
- getHours()** preuzima sat (0-23)
- getMilliseconds()** preuzima milisekunde (0-999)
- getMinutes()** preuzima minute (0-59)
- getMonth()** preuzima mesec (0-11)
- getSeconds()** preuzima sekunde (0-59)
- getTime()** preuzima vreme u ms od 1.1.1970.

dan je Fri Nov 24 2017 14:08:10 GMT+0100
(Central Europe Standard Time)

Prvi dan u sedmici je nedelja i to je broj 0

```
5 <p>Interni klok JavaScript počinje  
6 od ponoći 1.1.1970.<br>  
7 <b>getDate()</b> preuzima dan u me  
8 <b>getDay()</b> preuzima dan u ned  
9 <b>getFullYear()</b> preuzima godi  
10 <b>getHours()</b> preuzima sat (0  
11 <b>getMilliseconds()</b> preuzima  
12 <b>getMinutes()</b> preuzima minut  
13 <b>getMonth()</b> preuzima mesec (  
14 <b>getSeconds()</b> preuzima sekun  
15 <b>getTime()</b> preuzima vreme u  
16 </p>
```

```
17  
18 <p id="demo"></p>  
19  
20 <script>  
21 var dan = new Date();  
22 document.getElementById("demo").innerHTML =  
23 "dan je "+dan+  
24 "<br>dan u mesecu je "+dan.getDate()+  
25 "<br>dan u nedelji je "+dan.getDay()+  
26 "<br>godina je "+dan.getFullYear()+  
27 "<br>sati je "+dan.getHours()+  
28 "<br>milisekunde su "+dan.getMilliseconds()+  
29 "<br>minute su "+dan.getMinutes()+  
30 "<br>mesec je "+dan.getMonth()+  
31 "<br>sekunde su "+dan.getSeconds()+  
32 "<br>vreme je "+dan.getTime()+" ms";  
33 </script>
```

dan u mesecu je 24
dan u nedelji je 5
godina je 2017
sati je 14
milisekunde su 752
minute su 8
mesec je 10
sekunde su 10
vreme je 1511528890752 ms



JavaScript Date metode

- Set metode su korisne za postavljanje nekog dela datuma ili vremena (godina, mesec, dan, sat, minut, sekunda, ms)
- `setDate()` Postavi dan kao broj (1-31)
- `setFullYear()` Postavi godinu (opciono mesec i dan)
- `setHours()` Postavi sat (0-23)
- `setMilliseconds()` Postavi ms (0-999)
- `setMinutes()` Postavi minute (0-59)
- `setMonth()` Postavi mesec (0-11)
- `setSeconds()` Postavi sekunde (0-59)
- `setTime()` Postavi vreme (ms od 1.1.1970.)



primer16c.html

```

4 <p>
5 Set metode su korisne :
6 setDate() postavi dan
7 setFullYear() postavi
8 setHours() postavi sat
9 setMilliseconds() posta
10 setMinutes() postavi m
11 setMonth() postavi mes
12 setSeconds() postavi s
13 setTime() postavi vremen
14 <p id="demo"></p>
15 <script>
16 var dan0 = new Date();
17 var dan1 = new Date();
18 dan1.setFullYear(2020,
19 var dan2 = new Date();
20 dan2.setDate(15);
21 var dan3 = new Date();
22 dan3.setDate(dan3.getDate()
23 var dan4 = new Date();
24 dan4.setHours(21);
25 document.getElementById
26 "dan0 je, <b>" + dan0.to
27 "<br>postavi <b>FullYear
28 "<br>postavi <b>.setDate
29 "<br>postavi <b>.setHours(21)</b>, " + dan4.to
30 "<br>postavi <b>setDate(15)</b>, " + dan2.to
31 </script>
32 </body>

```

Set metode su korisne za postavljanje datuma i vremena.

setDate() postavi dan kao broj (1-31)

setFullYear() postavi godinu (opciono mesec i dan)

setHours() postavi sat (0-23)

setMilliseconds() postavi ms (0-999)

setMinutes() postavi minute (0-59)

setMonth() postavi mesec (0-11)

setSeconds() postavi sekunde (0-59)

setTime() postavi vreme (ms od 1.1.1970.)

Promene se
automatski
obrađuju
(31.2.2020
postaje
2.3.2020)

dan0 je, **Fri Nov 24 2017 15:18:25 GMT+0100 (Central Europe Standard Time)**

postavi **FullYear(2020,1,31)**, **Mon Mar 02 2020 15:18:25 GMT+0100 (Central Europe Standard Time)**

postavi **.setDate(...getDate() + 5)**, **Wed Nov 29 2017 15:18:25 GMT+0100 (Central Europe Standard Time)**

postavi **.setHours(21)**, **Fri Nov 24 2017 21:18:25 GMT+0100 (Central Europe Standard Time)**

postavi **setDate(15)**, **Wed Nov 15 2017 15:18:25 GMT+0100 (Central Europe Standard Time)**

Date metode da se postavi



JavaScript Date metode

- Date Input - Parsing Dates
- Ako postoji validan date string, `Date.parse()` metod se koristi za konverziju u milisekunde
- `Date.parse()` vraća broj ms između date i 1.1.1970.
- Nakon toga se broj milisekundi može konvertovati u date objekat



Date Input - Parsing Dates

Date.parse() vraća broj milisekundi između datuma i 1.1.1970.

za March 21, 2012
broj milisekundi je 1332284400000
ako se doda 99999 s, tada je
novi broj milisekundi 1332384399000
za dan Wed Mar 21 2012 00:00:00
GMT+0100 (Central Europe Standard
Time)
novi dan je Thu Mar 22 2012 03:46:39
GMT+0100 (Central Europe Standard
Time)

primer16d.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>Date.parse() vraća broj milisekun
5 između datuma i 1.1.1970.</p>
6 <p id="demo"></p>
7 <script>
8   var ms = Date.parse("March 21, 2012");
9   var ms2 = ms+99999*1000;
10  var dan = new Date(ms);
11  var dan2 = new Date(ms2);
12  document.getElementById("demo").innerHTML =
13  "za March 21, 2012<br> broj milisekundi je "+ms+
14  "<br>ako se doda 99999 s, tada je"+
15  "<br>novi broj milisekundi "+ms2+
16  "<br>za dan "+dan+
17  "<br>novi dan je "+ dan2;
18 </script>
19 </body>
20 </html>
```



Uporedjivanje datuma

- Datumi mogu lako da se upoređuju
- JavaScript broji mesece od 0 do 11
- Januar je 0
- Decembar je 11

Uporedjivanje datuma

danas je posle
Sat Jan 14 2017 16:58:42 GMT+0100
(Central Europe Standard Time)

danas je pre
Sun Jan 14 2018 17:00:08
GMT+0100 (Central Europe
Standard Time)

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p id="demo"></p>
5 <script>
6   var danas, zadataidan, tekst;
7   danas = new Date();
8   zadataidan = new Date();
9   zadataidan.setFullYear(2017, 0, 14);
10  if (zadataidan > danas) {
11    tekst = "danas je pre <br>";
12  } else {
13    tekst = "danas je posle <br>";
14  }
15  document.getElementById("demo").innerHTML =
16    tekst + zadataidan;
17 </script>
18 </body>
19 </html>
```



JavaScript Date metode

- za UTC vreme (Universal Time Zone dates)
 - `getUTCDate()` isto kao `getDate()`, ali vraća UTC datum
 - `getUTCDay()` isto kao `getDay()`, ali vraća UTC dan
 - `getUTCFullYear()` isto kao `getFullYear()`, ali vraća UTC godinu
 - `getUTCHours()` isto kao `getHours()`, ali vraća UTC čas
 - `getUTCMilliseconds()` isto kao `getMilliseconds()`, vraća UTC ms
 - `getUTCMinutes()` isto kao `getMinutes()`, ali vraća UTC minute
 - `getUTCMonth()` isto kao `getMonth()`, ali vraća UTC mesec
 - `getUTCSeconds()` isto kao `getSeconds()`, ali vraća UTC s



Nizovi

- JavaScript nizovi se koriste da se zapamti više podataka sa nazivom jedne promenljive

```
var nazivNiza = ["Nenad", "Marko", 24, "Zoran"];
```

- Blanko znak i prelazak u sledeći red nemaju uticaj
- Neki pregledači ne očekuju zarez posle poslednjeg elementa
- Ne koristiti **new**

```
var new nazivNiza = ["Nenad", "Marko", 24, "Zoran"];
```

- niz je tipa objekt,
- niz može sadržati objekte, funkcije, nizove



JavaScript nizovi

Ja imam 8 prijatelja:

Nenad,Marko,Branko,Milan,24,Zoran,Jovan,Dragan.

primer100.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <h4>JavaScript nizovi
5 <p id="demo"></p>
6 <script>
7   var mojiPrijatelji = ["Nenad", "Marko", "Branko",
8     "Milan", 24, "Zoran", "Jovan", "Dragan"];
9   document.getElementById("demo").innerHTML =
10  "Ja imam "+mojiPrijatelji.length+" prijatelja: "+
11  mojiPrijatelji+".";
12 </script>
13 </body>
14 </html>
```

JavaScript nizovi

Ja imam 8 prijatelja:

Nenad,Marko,Branko,Milan,24,Zoran,Jovan,Dragan.

Ako nema blanko karakter između navodnika i stringa, nema prelaz u sledeći red

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <h4>Tip JavaScript niza je object</h4>
5 <p id="demo"></p>
6 <script>
7   var mojiPrijatelji
8     " Nenad",
9     " Marko",
10    " Branko",
11    " Milan",
12    24,
13    " Zoran",
14    " Jovan",
15    " Dragan"
16   ];
17   document.getElementById("demo").innerHTML =
18   "Moji prijatelji su "+
19   mojiPrijatelji+", a tip je " +
20   typeof mojiPrijatelji;
21 </script>
22 </body>
23 </html>
```

Tip JavaScript niza je object

Moji prijatelji su Nenad, Marko, Branko, Milan, 24, Zoran, Jovan, Dragan, a tip je object

Ako ima blanko karakter između navodnika i stringa, ima prelaz u sledeći red


```

1 <!DOCTYPE html>
2 <html>
3 <body>
4   <h4>Niz mo
5   <p id="demo
6 <script>
7   function myFunction(a, b) {
8     return a * b;
9   }
10  var mojiPriatelji = [
11    " Nenad", " Marko",
12    " Jovan", " Dragan"
13  ];
14  var noviNiz= [];
15  noviNiz[0] = Date.now;
16  noviNiz[1] = myFunction;
17  noviNiz[2] = mojiPriatelji;
18  document.getElementById("demo").innerHTML =
19  noviNiz;
20 </script>
21 </body>
22 </html>

```

Niz može da ima objekat, metodu

function now() { [native code] }, function myFunction(a, b) { return a * b; }, Nenad, Marko, Jovan, Dragan





Osobine i metode niza

- Postoje ugrađene osobine i metode niza
- length, sort
- Broj elemenata niza za 1 je veći od najvećeg indeksa niza



primer16e.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <h4>metode length i sort za niz</h4>
5 <p id="demo"></p>
6 <script>
7   var mojiPriatelji = [
8     " Nenad", " Marko",
9     24, " Jovan", " Dragan"
10   ];
11   document.getElementById("demo").innerHTML =
12   "duzina niza je " + mojiPriatelji.length +
13   ", niz je<br>" + mojiPriatelji +
14   "<br>sortiran je<br>" + mojiPriatelji.sort();
15 </script>
16
17 </body>
18 </html>
```



primer16e.html



File | C:/aLutov

metode length i sort za niz

duzina niza je 5, niz je
Nenad, Marko,24, Jovan, Dragan,
sortiran je
Dragan, Jovan, Marko, Nenad,24



Petlje i nizovi

- Najbolje je da se koristi for petlja za nizove
- push za dodavanje elementa nizu
- Novi element se može dodati koristeći length



primer16f.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <h4>for loop i nenumerisana lista</h4>
5 <p id="demo"></p>
6 <script>
7   var mojiPrijatelji, tekst, nizLength, ind;
8   mojiPrijatelji = [
9     " Nenad", " Marko",
10    24, " Jovan", " Dragan"
11  ];
12   nizLength = mojiPrijatelji.length;
13   tekst = "<ul>";
14   for (ind = 0; ind < nizLength; ind++) {
15     tekst += "<li>" + mojiPrijatelji[ind] + "</li>";
16   }
17   tekst += "</ul>";
18   document.getElementById("demo").innerHTML = tekst;
19 </script>
20 </body>
21 </html>
```

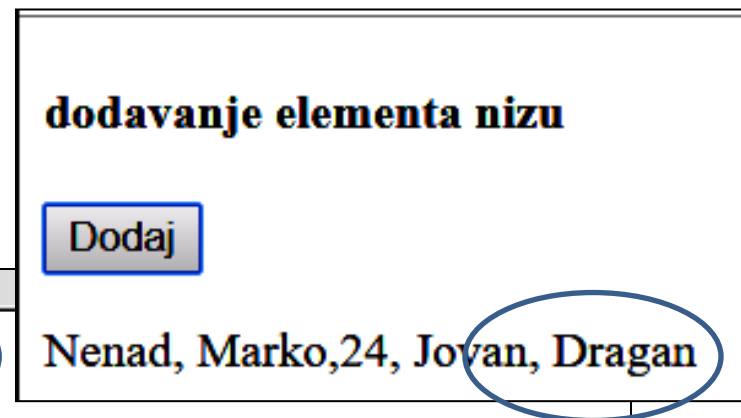
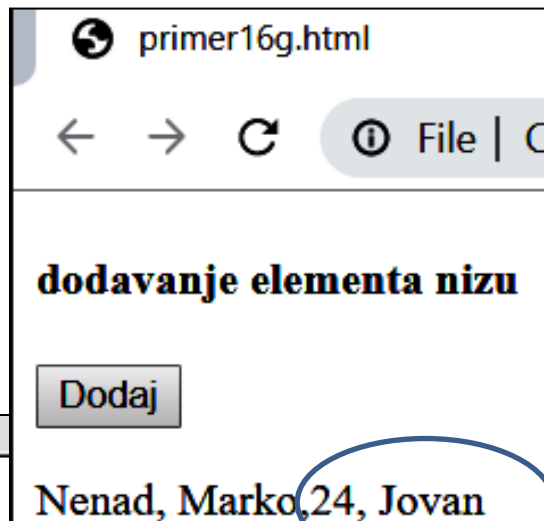
primer16f.html



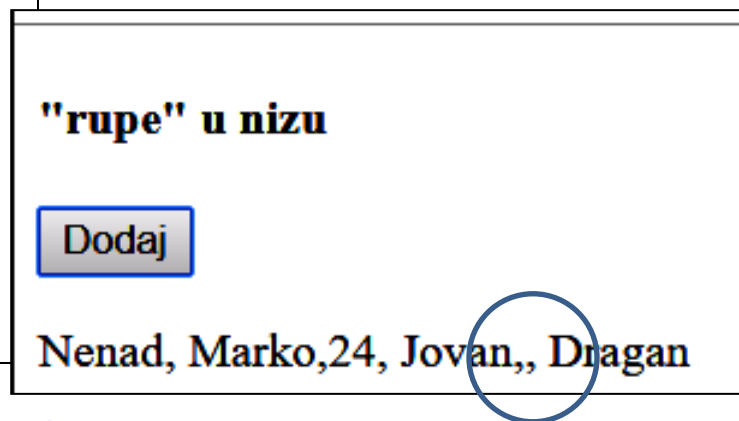
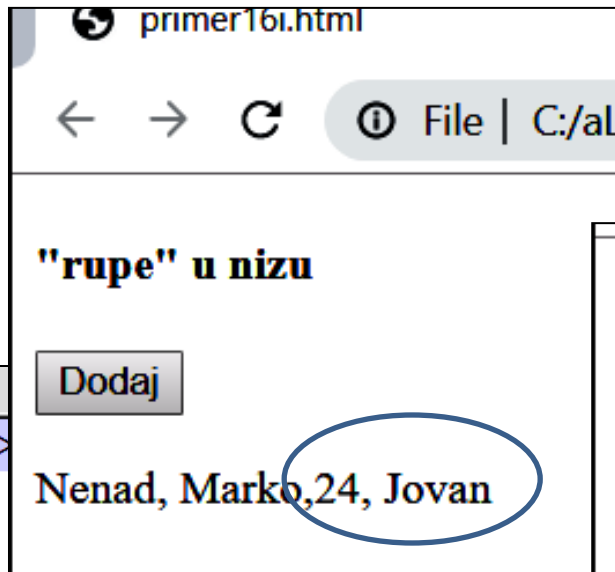
File | C:/aL

for loop i nenumerisana lista

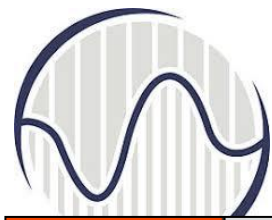
- Nenad
- Marko
- 24
- Jovan
- Dragan



```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <h4>dodavanje elementa nizu</h4>
5 <button onclick="myFunction()">Dodaj</button>
6 <p id="demo"></p>
7 <script>
8   mojiPrijatelji = [
9     " Nenad", " Marko",
10    24, " Jovan"];
11   noviPrijatelj = " Dragan";
12   document.getElementById("demo").innerHTML = mojiPrijatelji;
13   function myFunction() {
14     mojiPrijatelji.push(noviPrijatelj);
15     document.getElementById("demo").innerHTML = mojiPrijatelji;
16   }
17 </script>
18 </body>
19 </html>
```



```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <h4>"rupe" u nizu</h4>
5 <button onclick="myFunction()">Dodaj</button>
6 <p id="demo"></p>
7 <script>
8 mojiPriatelji = [
9     " Nenad", " Marko",
10    24, " Jovan"];
11 document.getElementById("demo").innerHTML = mojiPriatelji;
12 function myFunction() {
13     mojiPriatelji[1+mojiPriatelji.length]=" Dragan";
14     document.getElementById("demo").innerHTML = mojiPriatelji;
15 }
16 </script>
17 </body>
18 </html>
```



primer16j.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>"holes" in array</p>
5 <p id="demo"></p>
6 <script>
7   var mojiPrijetelji, tekst, nizLengt
8   mojiPrijetelji = [
9     " Nenad", " Marko",
10    24, " Jovan"];
11   mojiPrijetelji[6] = " Dragan";
12   nizLength = mojiPrijetelji.length;
13   tekst = "";
14   for (ind = 0; ind < nizLength; ind++) {
15     tekst += mojiPrijetelji[ind] + "<br>";
16   }
17   document.getElementById("demo").innerHTML = tekst;
18 </script>
19 </body>
20 </html>
```

primer16j.html



File

"holes" in array

Nenad

Marko

24

Jovan

undefined

undefined

Dragan



primer16k.html x

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>"holes" in array</p>
5 <p id="demo"></p>
6 <script>
7   var mojiPrijetelji, tekst, nizLength, ind;
8   mojiPrijetelji = [
9     " Nenad", " Marko",
10    24, " Jovan"];
11   mojiPrijetelji[6] = " Dragan";
12   nizLength = mojiPrijetelji.length;
13   tekst = "";
14   for (ind = 0; ind < nizLength; ind++) {
15     tekst += typeof mojiPrijetelji[ind] + "<br>";
16   }
17   document.getElementById("demo").innerHTML = tekst;
18 </script>
19 </body>
20 </html>
```

primer16k.html



"holes" in array

string

string

number

string

undefined

undefined

string



Asocijativni nizovi

- Mnogi programski jezici podržavaju imenovane indekse
- `var y = [ "blue", "red", "white" ];`
- `y[0]` je "blue"
- `y[1]` je "red"
- `var z = { color: "blue", fabric: "cotton" };`
- `z` je objekat , u drugim jezicima je asocijativni niz
- `z["color"]` je "blue"
- `z["fabric"]` je "cotton"
- za `y` pridružujemo broj elementu niza,
za `z` pridružujemo ključno ime elementa



Asocijativni nizovi

- Nizovi sa asocijativnim imenima indeksa nazivaju se asocijativni nizovi (associative arrays, hashes)
- JS ne podržava nizove sa imenovanim indeksima
- JavaScript nizovi uvek koriste numerisane indekse
- Ako se koriste imenovani indeksi, JS će redefinisati niz kao standardni objekat; posle toga, neke metode i osobine mogu da proizvedu neočekivan rezultat



```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>imenovani nizovi JS postaju
5 <p id="demo"></p>
6 <script>
7   var osoba1 = [];
8   var osoba2 = [];
9   osoba1["ime"] = "Marko";
10  osoba1["prezime"] = "Markov";
11  osoba1["godine"] = 23;
12  osoba2[0] = " Marko";
13  osoba2[1] = " Markov";
14  osoba2[2] = 23;
15  document.getElementById("demo").innerHTML =
16  osoba2 + "<br> " +
17  osoba1[0] + " " + osoba1.length + "<br> " +
18  osoba1["ime"] + ", " + osoba1["prezime"] +
19  ", " + osoba1["godine"] + "<br> " +
20  osoba2[0] + " " + osoba2.length;
21 </script>
22 </body>
23 </html>
```

primer16m.html

← → ↻ ⓘ File | C:/aLutovac

imenovani nizovi JS postaju objekti

Marko, Markov,23

undefined 0

Marko, Markov, 23

Marko 3



Razlike između nizova i objekata

- JS koristi brojne indekse za nizove
- JS koristi imenovane indekse za objekte
- Nizovi su specijalna vrsta objekata sa brojnim indeksima
- JS ne podržava asocijativne indekse za nizove; ako je potrebno da se koriste asocijativni indeksi, treba da se koriste objekti
- ako je potrebno da se koriste brojni indeksi, treba da se koriste nizovi



primer16n.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>nizovi ili objekti</p>
5 <p id="demo"></p>
6 <script>
7   var osoba1 = [];
8   var osoba2 = [];
9   osoba1["ime"] = "Marko";
10  osoba1["prezime"] = "Markov";
11  osoba1["godine"] = 23;
12  osoba2[0] = " Marko";
13  osoba2[1] = " Markov";
14  osoba2[2] = 23;
15  document.getElementById("demo").innerHTML =
16  osoba1+"ne ispisuje nista "+typeof osoba1+"<br>"+
17  " ispisuje "+osoba1.ime+" "+typeof osoba1.ime+
18  "<br>" + osoba2 + " " + typeof osoba2;
19 </script>
20 </body>
21 </html>
```

primer16n.html



File | C

nizovi ili objekti

ne ispisuje nista object
ispisuje Marko string
Marko, Markov,23 object

JS operator typeof za niz vraća "object" !!!



primer16p.html

Array.isArray

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>nizovi ili objekti</p>
5 <p id="demo"></p>
6 <script>
7   var osoba1 = [];
8   var osoba2 = [];
9   osoba1["ime"] = "Marko";
10  osoba1["prezime"] = "Markov";
11  osoba1["godine"] = 23;
12  osoba2[0] = " Marko";
13  osoba2[1] = " Markov";
14  osoba2[2] = 23;
15  document.getElementById("demo").innerHTML =
16  "osoba1 " + Array.isArray(osoba1) + "<br>" +
17  "osoba2 " + Array.isArray(osoba2) ;
18 </script>
19 </body>
20 </html>
```

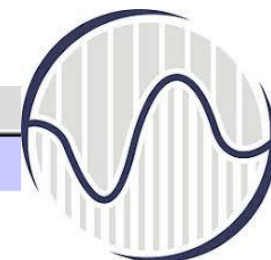
primer16p.html



nizovi ili objekti

osoba1 true

osoba2 true



primer16r.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>nizovi ili objekti</p>
5 <p id="demo"></p>
6 <script>
7   var osoba1 = [];
8   var osoba2 = [];
9   var osoba3 = " Marko";
10  osoba1["ime"] = "Marko"; osoba1["godine"] = 23;
11  osoba1["prezime"] = "Markov";
12  osoba2[0] = " Marko";
13  osoba2[1] = " Markov"; osoba2[2] = 23;
14  document.getElementById("demo").innerHTML =
15  "osoba1 "+ isArray(osoba1)+"<br>"+
16  "osoba2 "+ isArray(osoba2)+"<br>"+
17  "osoba3 "+ isArray(osoba3);
18  function isArray(myArray) {
19    return myArray.constructor.toString().indexOf("Array") > -1;
20  }
21 </script>
22 </body>
23 </html>
```

primer16r.html



nizovi ili objekti

osoba1 true

osoba2 true

osoba3 false

constructor.toString().indexOf



primer16s.html

instanceof Array

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>nizovi ili objekti</p>
5 <p id="demo"></p>
6 <script>
7   var osoba1 = [];
8   var osoba2 = [];
9   var osoba3 = " Marko";
10  osoba1["ime"] = "Marko"; osoba1["godine"] = 23;
11  osoba1["prezime"] = "Markov";
12  osoba2[0] = " Marko";
13  osoba2[1] = " Markov"; osoba2[2] = 23;
14  document.getElementById("demo").innerHTML =
15  "osoba1 " + (osoba1 instanceof Array) + "<br>" +
16  "osoba2 " + (osoba2 instanceof Array) + "<br>" +
17  "osoba3 " + (osoba3 instanceof Array) ;
18 </script>
19 </body>
20 </html>
```

primer16s.html



nizovi ili objekti

osoba1 true

osoba2 true

osoba3 false



JavaScript metode za nizove

- metoda `toString()` konvertuje niz u string (vrednosti niza su razdvojeni zarezima)
- metoda `join()` spaja sve elemente niza u string slično kao `toString()`, može da se definiše separator



primer16t.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <h4>toString() metoda</h4>
5 <p id="demo"></p>
6 <script>
7     mojiPrijatelji = [
8         " Nenad", " Marko", 24, " Jovan"];
9     document.getElementById("demo").innerHTML =
10     mojiPrijatelji.toString()+"<br>"+
11     "typeof mojiPrijatelji.toString() je "+
12     typeof mojiPrijatelji.toString()+"<br>"+
13     "typeof mojiPrijatelji je "+
14     typeof mojiPrijatelji;
15 </script>
16 </body>
17 </html>
```

primer16t.html

← → ↻ ⓘ File | C:/aLutovac/20

toString() metoda

Nenad, Marko,24, Jovan
typeof mojiPrijatelji.toString() je string
typeof mojiPrijatelji je object



primer16t2.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <h4>join() metoda</h4>
5 <p id="demo"></p>
6 <script>
7   mojiPrijatelji = [
8     "Nenad", "Marko", 24, "Jovan"];
9   document.getElementById("demo").innerHTML =
10  mojiPrijatelji.join(" * ") + "<br>" +
11  'typeof mojiPrijatelji.join(" * ") je ' +
12  'typeof (mojiPrijatelji.join(" * ")) + "<br>" +
13  'typeof mojiPrijatelji je ' +
14  'typeof mojiPrijatelji;
15 </script>
16 </body>
17 </html>
```

primer16t2.html

< > ↻ ⓘ File | C:/aLutovac/201

join() metoda

Nenad * Marko * 24 * Jovan
typeof mojiPrijatelji.join(" * ") je string
typeof mojiPrijatelji je object



JavaScript Popping, Pushing

- Jednostavno se eliminiše element niza
- Jednostavno se dodaje novi element niza
- popping, pushing
- popping out, pushing into array
- **pop() metoda odstranjuje** poslednji element niza
- pop() metoda **vraća element** koji je odstranjen
- **push() metod dodaje novi** element na kraj niza
- **push() metod vraća dužinu novog niza**



primer16u.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <h2>pop() </h2>
5 <p id="demo1"></p>
6 <p id="demo2"></p>
7 <p id="demo3"></p>
8 <script>
9   var mojiPriatelji = [" Nenad",24," Marko"," Jovan"];
10  document.getElementById("demo1").innerHTML =
11  "originalni niz je : " + mojiPriatelji;
12  document.getElementById("demo2").innerHTML =
13  "eliminisan je : " + mojiPriatelji.pop();
14  document.getElementById("demo3").innerHTML =
15  "bez zadnjeg elementa je : " + mojiPriatelji;
16 </script>
17 </body>
18 </html>
```

primer16u.html

← → ↻ ⓘ File | C:/aLutovac/2019/IF

pop()

originalni niz je : Nenad,24, Marko, Jovan

eliminisan je : Jovan

bez zadnjeg elementa je : Nenad,24, Marko



primer16v.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4   <h2>pop() </h2>
5   <p id="demo1"></p>
6   <p id="demo2"></p>
7   <p id="demo3"></p>
8   <script>
9     var mojiPriatelji = [" Nenad",24," Marko"," Jovan"];
10    document.getElementById("demo1").innerHTML =
11    "originalni niz je : " + mojiPriatelji;
12    document.getElementById("demo2").innerHTML =
13    "eliminisan je : " + mojiPriatelji.pop();
14    document.getElementById("demo3").innerHTML =
15    "bez zadnjeg elementa je : " + mojiPriatelji;
16  </script>
17 </body>
18 </html>
```

primer16v.html

x

+



File | C:/aLutovac/2019/I

pop()

originalni niz je : Nenad,24, Marko, Jovan

eliminisan je : Jovan

bez zadnjeg elementa je : Nenad,24, Marko



primer16v2.html

1 <!DOCTYPE html>

2 <html>

3 <body>

4 <p>push() metod</p>

5

6 <button onclick="myFunction()">Ubaci u niz</button>

7 <p id="demo1"></p>

8 <p id="demo2"></p>

9 <script>

10 var mojiPrijatelji = [" Nenad",24," Marko"];

11 document.getElementById("demo1").innerHTML = mojiPrijatelji;

12 function myFunction() {

13 document.getElementById("demo2").innerHTML =

14 mojiPrijatelji.push("Jovan");

15 document.getElementById("demo1").innerHTML =

16 mojiPrijatelji;

17 }

18 </script>

19 </body>

20 </html>

primer16v2.html

← → ↻ ⓘ

push() metod

Ubaci u niz

Nenad,24, Marko

push() metod

Ubaci u niz

Nenad,24, Marko,Jovan

4

push() metod

Ubaci u niz

Nenad,24, Marko,Jovan,Jovan

5



JavaScript Shifting

- Shifting je ekvivalentno popping, samo ya prvi umesto ya poslednji element
- `shift()` metoda eliminiše prvi element i pomera ostale za jedan indeks ka početku
- `shift()` metoda vraća string koji je eliminisan
- `unshift()` metoda dodaje novi element nizu na početku
- `unshift()` metod vraća dužinu novog niza



primer16u3.html

← → ↻ ⓘ File | C:/aLutovac/2019/

shift()

originalni niz je : Nenad, Marko,24, Jovan

eliminisan je : Nenad

bez prvog elementa je : Marko,24, Jovan

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <h2>shift()</h2>
5 <p id="demo1"></p>
6 <p id="demo2"></p>
7 <p id="demo3"></p>
8 <script>
9 var mojiPriatelji = ["Nenad","Marko",24,"Jovan"];
10 document.getElementById("demo1").innerHTML =
11 "originalni niz je : " + mojiPriatelji;
12 document.getElementById("demo2").innerHTML =
13 "eliminisan je : " + mojiPriatelji.shift();
14 document.getElementById("demo3").innerHTML =
15 "bez prvog elementa je : " + mojiPriatelji;
16 </script>
17 </body>
18 </html>
```



primer16v3.html

unshift() metod

Ubaci u niz

Nenad,24, Marko

unshift() metod

Ubaci u niz

Jovan, Nenad,24, Marko

4

unshift() metod

Ubaci u niz

Jovan,Jovan, Nenad,24, Marko

5

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>unshift() metod</p>
5
6 <button onclick="myFunction()">Ubaci u niz</button>
7 <p id="demo1"></p>
8 <p id="demo2"></p>
9 <script>
10 var mojiPrijetelji = ["Nenad",24,"Marko"];
11 document.getElementById("demo1").innerHTML = mojiPrijetelji;
12 function myFunction() {
13     document.getElementById("demo2").innerHTML =
14     mojiPrijetelji.unshift("Jovan");
15     document.getElementById("demo1").innerHTML =
16     mojiPrijetelji;
17 }
18 </script>
19 </body>
20 </html>
```

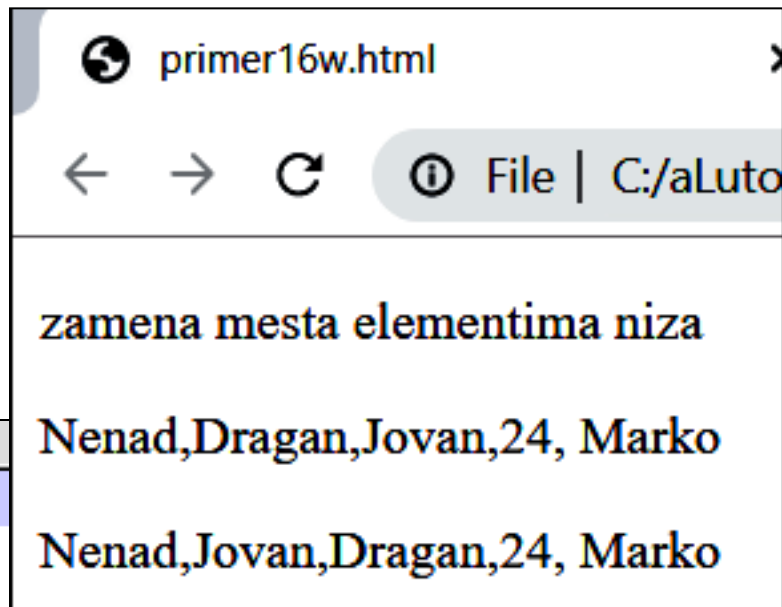
VISER 2019 NRT i IS plan 2017

40



Zamena elemenata niza

- Elementima niza se pristupa preko indeksa
- Početni indeks niza je 0
- [0] je prvi element niza, [1] je drugi, [2] je treći ...



```
primer16w.html x
1  <!DOCTYPE html>
2  <html>
3  <body>
4  <p>zamena mesta elementima niza</p>
5  <p id="demo1"></p>
6  <p id="demo2"></p>
7  <script>
8  var mojiPriatelji = [" Nenad","Dragan","Jovan",24," Marko"];
9  document.getElementById("demo1").innerHTML = mojiPriatelji;
10 var x = mojiPriatelji[1] ;
11 mojiPriatelji[1] = mojiPriatelji[2];
12 mojiPriatelji[2] = x;
13 document.getElementById("demo2").innerHTML = mojiPriatelji;
14 </script>
15 </body>
16 </html>
```



Brisanje elemenata niza

- JavaScript operator delete
- delete može da napravi rupu u nizu



primer16w2.html

← → ↻ ⓘ File | C:/aLut

brisanje elementa niza

Nenad,Dragan,Jovan,24, Marko

Nenad,,Jovan,24, Marko

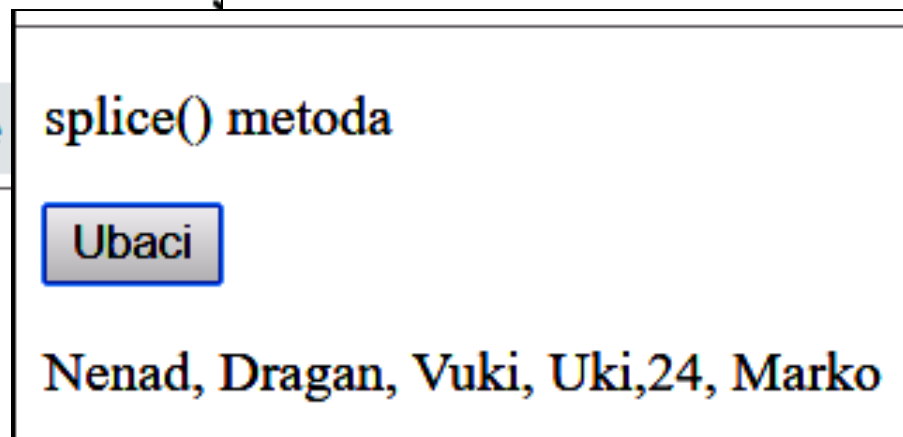
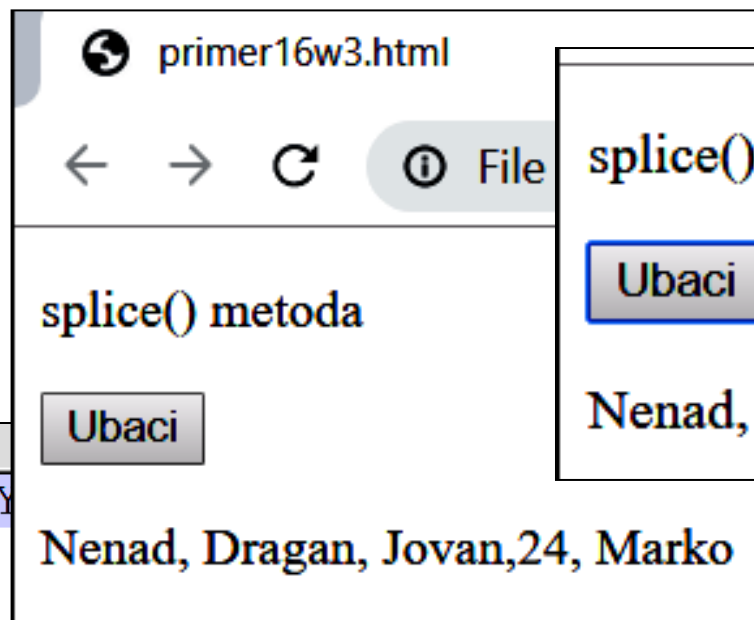
primer16w2.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>brisanje elementa niza</p>
5 <p id="demo1"></p>
6 <p id="demo2"></p>
7 <script>
8   var mojiPrijatelji = [" Nenad","Dragan","Jovan",24," Marko"];
9   document.getElementById("demo1").innerHTML = mojiPrijatelji;
10  var x = mojiPrijatelji[1] ;
11  delete mojiPrijatelji[1];
12  document.getElementById("demo2").innerHTML = mojiPrijatelji;
13 </script>
14 </body>
15 </html>
```




Dodavanje i brisanje elemenata niza

- splice() metod za dodavanje novih elementa niza
- Prvi parametar određuje poziciju gde novi element treba da se doda
- Drugi parametar određuje koliko elemenata treba izbaciti
- Ostali parametri definišu nove elemente koje treba dodati nizu



```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>splice() metoda</p>
5 <button onclick="myFunction()">Ubaci</button>
6 <p id="demo"></p>
7 <script>
8   var noviPriatelji = [" Vuki"," Uki"];
9   var mojiPriatelji = [" Nenad"," Dragan"," Jovan",24," Marko"];
10  document.getElementById("demo").innerHTML = mojiPriatelji;
11  function myFunction() {
12    mojiPriatelji.splice(2, 1, noviPriatelji);
13    document.getElementById("demo").innerHTML = mojiPriatelji;
14  }
15 </script>
16 </body>
17 </html>
```



Brisanje elemenata niza bez rupa

- splice() metod za brisanje elementa niza bez rupa
- Prvi parametar određuje poziciju gde novi element treba da se doda
- Drugi parametar određuje koliko elemenata treba izbaciti (obrisati)
- Ostali parametri definišu nove elemente koje treba dodati nizu, ako se izostave, neće biti novih elemenata



splice() metoda

Obrisi

Nenad, Dragan, Jovan,24, Marko

splice() metoda

Obrisi

Nenad, Dragan,24, Marko

splice() metoda

Obrisi

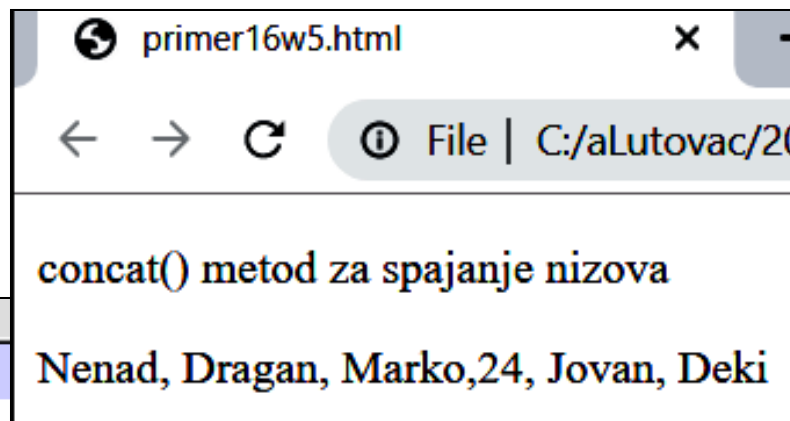
Nenad, Dragan, Marko

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>splice() metoda</p>
5 <button onclick="myFunction()">Obrisi</button>
6 <p id="demo"></p>
7 <script>
8 var mojiPrijatelji = [" Nenad"," Dragan"," Jovan",24," Marko"];
9 document.getElementById("demo").innerHTML = mojiPrijatelji;
10 function myFunction() {
11     mojiPrijatelji.splice(2, 1);
12     document.getElementById("demo").innerHTML = mojiPrijatelji;
13 }
14 </script>
15 </body>
16 </html>
```



Spajanje nizova

- `concat()` metod pravi novi niz spajanjem postojećih nizova
- `concat()` metod uvek pravi novi niz, ne menja postojeće
- `concat()` metod može da spaja više nizova
- `concat()` metod može da dodaje vrednosti



```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>concat() metod za spajanje nizova</p>
5 <p id="demo"></p>
6 <script>
7   var mojiPrijatelji = [" Nenad"," Dragan"," Marko"];
8   var noviPrijatelji = [" Jovan"," Deki"];
9   var x = 24;
10  var sviPrijatelji = mojiPrijatelji.concat(x, noviPrijatelji);
11  document.getElementById("demo").innerHTML = sviPrijatelji;
12 </script>
13 </body>
14 </html>
```



Slicing

- `slice()` metod izdvaja delove niza u novi niz
- `slice()` metod pravi novi niz
ne menja postojeće nizove



primer16w6.html

← → ↻ ⓘ File | C:/aLutovac

slice()

Nenad, Dragan, Marko, Jovan, Deki

Marko, Jovan, Deki

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>slice()</p>
5 <p id="demo"></p>
6 <script>
7   var mojiPriatelji = [" Nenad"," Dragan"," Marko",
8                       " Jovan"," Deki"];
9   var noviPriatelji = mojiPriatelji.slice(2);
10  document.getElementById("demo").innerHTML =
11  mojiPriatelji + "<br><br>" + noviPriatelji;
12 </script>
13 </body>
14 </html>
```



slice()

Nenad, Dragan, Marko, Jovan, Deki, Uki
Marko, Jovan

primer16w7.html x

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>slice()</p>
5 <p id="demo"></p>
6 <script>
7   var mojiPrijatelji = [" Nenad"," Dragan"," Marko",
8                         " Jovan"," Deki"," Uki"];
9   var noviPrijatelji = mojiPrijatelji.slice(2,4);
10  document.getElementById("demo").innerHTML =
11  mojiPrijatelji + "<br>" + noviPrijatelji;
12 </script>
13 </body>
14 </html>
```



toString() metoda - opšta

- toString() metoda vraća broj kao string
- Sve methods za brojeve mogu da se koriste za bilo koji tip brojeva (literali, promenljive, izrazi)
- Literal je notacija za predstavljanje fiksne vrednosti u izvornom kodu
- Literali se koriste za inicijalizaciju promenljivih

```
var x = 123;
```

```
x.toString();           // returns 123 from variable x
```

```
(123).toString();       // returns 123 from literal 123
```

```
(100+23).toString();    // returns 123 from expression 100+23
```



toString() metoda za nizove

- Svi JavaScript objekti imaju toString() metod
- JS po automatizmu konvertuje niz u string kada se očekuju primitivne vrednosti
- To se uvek dešava kada treba da se dobije izlazna vrednost niza



sort() i reverse() metode

- sort() metoda sortira niz po azbučnom redu
- reverse() metoda sortira u obrnutom redosledu
- reverse() metoda se koristi za sortiranje po opadajućem redosledu



sort() metoda

Sortiraj

Nenad, Dragan, Marko, 24, 101,
Jovan, Deki, Uki

sort() metoda

Sortiraj

Deki, Dragan, Jovan, Marko,
Nenad, Uki, 101, 24

primer16x.html

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>sort() metoda</p>
5 <button onclick="myFunction()">Sortiraj</button>
6 <p id="demo"></p>
7 <script>
8   var mojiPrijatelji = [" Nenad", " Dragan", " Marko",
9     24, 101, " Jovan", " Deki", " Uki"];
10  document.getElementById("demo").innerHTML = mojiPrijatelji;
11  function myFunction() {
12    mojiPrijatelji.sort();
13    document.getElementById("demo").innerHTML = mojiPrijatelji;
14  }
15 </script>
16 </body>
17 </html>
```

Kada se sortiraju brojevi, "25" je veće od "101", zato što je "2" veće od "1"



Kada se sortiraju brojevi, "25" je veće od "100", zato što je "2" veće od "1"

reverse() metoda

Sortiraj

Nenad, Dragan, Marko, 24, 101,
Jovan, Deki, Uki

reverse() metoda

Sortiraj

Uki, Deki, Jovan, 101, 24, Marko,
Dragan, Nenad

primer16x2.html

```
1 <!DOC
2 <html
3 <body>
4 <p>reverse() metoda</p>
5 <button onclick="myFunction()">Sortiraj</button>
6 <p id="demo"></p>
7 <script>
8 var mojiPrijatelji = [" Nenad"," Dragan"," Marko",
9     24,101," Jovan"," Deki"," Uki"];
10 document.getElementById("demo").innerHTML = mojiPrijatelji;
11 function myFunction() {
12     mojiPrijatelji.reverse();
13     document.getElementById("demo").innerHTML = mojiPrijatelji;
14 }
15 </script>
16 </body>
17 </html>
```

Sortiranje brojeva predstavljenih kao stringovi može da ne da tačan rezultat



Sortiraj

40,100,1,5,25,10

Sortiraj

1,5,10,25,40,100

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <button onclick="myFunction()">Sortiraj</button>
5 <p id="demo"></p>
6 <script>
7   var mojiPrijetelji = [40, 100, 1, "Marko", 5, 25, 10];
8   var koordinate = [40, 100, 1, 5, 25, 10];
9   document.getElementById("demo").innerHTML = koordinate;
10  function myFunction() {
11      koordinate.sort(function(a, b){return a - b});
12      document.getElementById("demo").innerHTML = koordinate;
13  }
14 </script>
15 </body>
16 </html>
```




Sortiraj

40,100,1,Marko,5,25,10

Sortiraj

1,40,100,Marko,5,10,25

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <button onclick="myFunction()">Sortiraj</button>
5 <p id="demo"></p>
6 <script>
7   var mojiPrijetelji = [40, 100, 1, "Marko", 5, 25, 10];
8   var koordinate = [40, 100, 1, "Marko", 5, 25, 10];
9   document.getElementById("demo").innerHTML = koordinate;
10  function myFunction() {
11      koordinate.sort(function(a, b){return a - b});
12      document.getElementById("demo").innerHTML = koordinate;
13  }
14 </script>
15 </body>
16 </html>
```

Ne radi za string umetnut u niz brojeva



Sortiraj

40,100,1,5,25,10

Sortiraj

100,40,25,10,5,1

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <button onclick="myFunction()">Sortiraj</button>
5 <p id="demo"></p>
6 <script>
7   var koordinate = [40, 100, 1, 5, 25, 10];
8   document.getElementById("demo").innerHTML = koordinate;
9   function myFunction() {
10     koordinate.sort(function(a, b){return b - a});
11     document.getElementById("demo").innerHTML = koordinate;
12   }
13 </script>
14 </body>
15 </html>
```



Sortiraj

40,100,1,Marko,5,25,10

Sortiraj

100,40,1,Marko,25,10,5

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <button onclick="myFunction()">Sortiraj</button>
5 <p id="demo"></p>
6 <script>
7   var mojiPrijetelji = [40, 100, 1, "Marko", 5, 25, 10];
8   var koordinate = [40, 100, 1, "Marko", 5, 25, 10];
9   document.getElementById("demo").innerHTML = koordinate;
10  function myFunction() {
11      koordinate.sort(function(a, b){return b - a});
12      document.getElementById("demo").innerHTML = koordinate;
13  }
14 </script>
15 </body>
16 </html>
```

Ne radi za string umetnut u niz brojeva



Funkcija za upoređivanje

- Funkcija za upoređivanje se definiše za potrebe alternativnog sortiranja
- Funkcija za upoređivanje će vratiti negativnu, nultu ili pozitivnu vrednost, u zavisnosti od argumenta
- Kada `sort()` funkcija upoređuje dve vrednosti, ona šalje vrednosti funkciji za upoređivanje, pa sortira vrednosti prema vrednosti koja je vraćena (negativna, nulta, pozitivna)


```

1  <!DOCTYPE html>
2  <html>
3  <body>
4  <button onclick="myFunction1()">Sortiraj stringove</button>
5  <button onclick="myFunction2()">Sortiraj brojeve</button>
6  <p id="demo"></p>
7  <script>
8  var mojiPrijatelji = [" Nenad"," Dragan"," Marko",
9      24,101," Jovan"," Deki"," Uki"];
10 var koordinate = [40, 100, 1, " Marko", 25, 10];
11 document.getElementById("demo").innerHTML = koordinate;
12 function myFunction1() {
13     koordinate.sort();
14     document.getElementById("demo").innerHTML = koordinate;
15 }
16 function myFunction2() {
17     koordinate.sort(function(a, b){return a - b});
18     document.getElementById("demo").innerHTML = koordinate;
19 }
20 </script>
21 </body>
22 </html>

```

Sortiraj stringove

Sortiraj stringove

Sortiraj brojeve

40,100,1, Marko,25,10

Sortiraj stringove

Sortiraj brojeve

Marko,1,10,100,25,40

Sortiraj stringove

Sortiraj brojeve

Marko,1,10,25,40,100

Marko,1,10,100,25,40

Sortiraj brojeve

Sortiraj stringove

Sortiraj brojeve

40,100,1, Marko,25,10

Sortiraj stringove

Sortiraj brojeve

1,40,100, Marko,10,25

Sortiraj stringove

Sortiraj brojeve

Marko,1,10,100,25,40

Sortiraj stringove

Sortiraj brojeve

Marko,1,10,25,40,100



Sortiranje nizova po slučajnom redosledu

```
var points = [40, 100, 1, 5, 25, 10];  
points.sort(function(a, b){return 0.5 - Math.random()});
```




Sortiraj 40,100,1,5,25,10	Sortiraj 100,40,1,5,25,10	Sortiraj 25,100,40,1,5,10
---	---	---

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <button onclick="myFunction()">Sortiraj</button>
5 <p id="demo"></p>
6 <script>
7   var koordinate = [40, 100, 1, 5, 25, 10];
8   document.getElementById("demo").innerHTML = koordinate;
9   function myFunction() {
10     koordinate.sort(function(a, b){return 0.5 - Math.random()});
11     document.getElementById("demo").innerHTML = koordinate;
12   }
13 </script>
14 </body>
15 </html>
```



Pronaći najveći, najmanju, vrednost niza

- Ne postoji ugrađena funkcija za nalaženje najveće ili najmanje vrednosti niza
- Posle sortiranja niza, može se iskoristiti indeks za dobijanje najveće ili najmanje vrednosti
- Sortiranje celog niza je neefikasan metod za nalaženje najveće ili najmanje vrednosti



Math.max(), Math.min() za niz

- Za potrebe nalaženja najveće ili najmanje vrednosti niza
- Mogu se napisati sopstveni skriptovi kao metode korišćenjem petlji



Sortiranje nizova sa objektima

- `sort()` metoda može biti efikasna u ovim slučajevima
- Napisati sopstvenu funkciju za upoređivanje osobina objekata



```
4 <button onclick="myFunction()">Sortiraj</button>
5 <p id="demo"></p>
6 <script>
7   var auto = [
8     {tip:"Fijat", godina:2016},
9     {tip:"Golf", godina:2001},
10    {tip:"Audi", godina:2010}]
11
12   displayAuto();
13
14   function myFunction() {
15     auto.sort(function(a, b){return a.godina - b.godina});
16     displayAuto();
17   }
18   function displayAuto() {
19     document.getElementById("demo").innerHTML =
20     auto[0].tip + " " + auto[0].godina + "<br>" +
21     auto[1].tip + " " + auto[1].godina + "<br>" +
22     auto[2].tip + " " + auto[2].godina;
23   }
24 </script>
```

Sortiraj

Fijat 2016
Golf 2001
Audi 2010

Sortiraj

Golf 2001
Audi 2010
Fijat 2016


```

4 <button onclick="myFunction()">Sortiraj</button>
5 <p id="demo"></p>
6 <script>
7 var auto = [
8   {tip:"Fijat", godina:2016},
9   {tip:"Golf", godina:2001},
10  {tip:"Audi", godina:2010}]
11 displayAuto();
12 function myFunction() {
13   auto.sort(function(a, b){
14     var x = a.tip.toLowerCase();
15     var y = b.tip.toLowerCase();
16     if (x < y) {return -1;}
17     if (x > y) {return 1;}
18     return 0;
19   });
20   displayAuto();
21 }
22 function displayAuto() {
23   document.getElementById("demo").innerHTML =
24   auto[0].tip + " " + auto[0].godina + "<br>" +
25   auto[1].tip + " " + auto[1].godina + "<br>" +
26   auto[2].tip + " " + auto[2].godina;
27 }
28 </script>

```

Sortiraj

Fijat 2016
Golf 2001
Audi 2010

Sortiraj

Audi 2010
Fijat 2016
Golf 2001