CENTAR ZA
mlade talente

Veb programiranje

Tema: JS - Nizovi

JS Nizovi

- Nizovi služe da uskladište više vrednosti u jednu promenljivu

```
var niz = ["jabuka", "banana", "trešnja", "kivi", "jagoda"];
```

Šta je niz?

- Niz je specijalna promenljiva koja može da sadrži više od jedne vrednosti

```
var ime_niza= [element1, element2, ...];
```

- Niz sadrži više vrednosti pod jednim imenom
- Vrednostima pristupamo pomoću rednog broja - indeksa
- Kreiranje novog praznog niza:

```
var ime_niza= [];
```

Pristup elementima niza

- Pomoću rednog broja - indeksa

```
var voce = niz[0];
```

- Izmena vrednodti element niza

```
var niz = ["jabuka", "banana", "trešnja", "kivi", "jagoda"];
niz[0] = "krompir";
```

- **Pažnja**: prvi elemenat niza ima indeks 0, drugi elemenat niza ima index 1, itd.

Metode i osobine niza

- Osobina length vraća broj elemenata u nizu

```
var niz = ["jabuka", "banana", "trešnja", "kivi", "jagoda"];

var duz = niz.length;  // dužina liste je 5
```

- **sort()** vraća sortiran niz

```
var nizS = niz.sort();
```

Obilazak elemenata niza

- Da obiđemo sve elemente niza koristimo petlje

```
for (i = 0; i < niz.length ; i++) {
  //***pristupamo svakom elementu niza niz[i]**
}
```

- Možemo koristiti i funkciju **forEach():**

```
niz.forEach(nekaFunkcija);

function nekaFunkcija(parametar) {
//pristupamo svakom elementu niza koji se nalazi u value
}
```

Dodavanje elementa u niz

- Da dodamo elemenat na kraj niza koristimo metod **push()**

```
var niz = ["jabuka", "banana", "trešnja", "kivi", "jagoda"];
niz.push("limun");
```

- **push()** vraća dužinu novog niza:

```
var x = fruits.push("mango");
```

- A možemo i bez **push()**:

```
niz[niz.length] = "limun"; // stavlja limun na kraj niza
```

Konvertovanje nizova u string

- Konvertovanje niza u string, sa elementima razdvojenim zarezima **toString():**

```
var niz = ["jabuka", "banana", "trešnja", "kivi", "jagoda"];
var novistring = niz.toString();
```

- Metod **`join()`** takođe spaja sve element niza u string, ali možemo da biramo separator**:**

```
var novistring = niz.join(" * ");
```

Uklanjanje elemenata iz niza

- **pop()** metoda uklanja poslednji element iz niza:

```
var niz = ["jabuka", "banana", "trešnja", "kivi", "jagoda"];
niz.pop();// izbacuje elemenat "jagoda"
```

- **pop()** metoda može i da vrati vrednost izbačenog elementa niza:

```
var niz = ["jabuka", "banana", "trešnja", "kivi", "jagoda"];
var x = niz.pop();// vrednost u x je "jagoda"
```

Uklanjanje elemenata iz niza

- **shift()** metoda uklanja **prvi** element iz niza, i pomera sve ostale element za jedno mesto u levu stranu:

```
var niz = ["jabuka", "banana", "trešnja", "kivi", "jagoda"];
niz.shift();// izbacuje elemenat "jabuka"
```

- **shift()** metoda vraća vrednost izbačenog elementa niza:

```
var niz = ["jabuka", "banana", "trešnja", "kivi", "jagoda"];
var x = niz.shift();// vrednost u x je "jabuka"
```

Brisanje elemenata iz niza

- **delete** operator briše elemenat iz niza:

```
var niz = ["jabuka", "banana", "trešnja", "kivi", "jagoda"];
delete niz[1];// menja elemenat sa indexom 1 u undefined
```

- **delete** ostavlja undefined rupe u nizu, pa je bolje koristiti **pop()** i **shift()**

Dodavanje elemenata u sredinu niza

- **splice()** metod dodaje elemenat u niz

```
var niz = ["jabuka", "banana", "trešnja", "kivi",
"jagoda"];
niz.splice(2,0, "limun", "kruska")
```

- Prvi parametar određuje poziciju na koju se dodaju elementi
- Drugi parametar određuje koliko elemenata treba ukloniti
- Ostali paramatri su elementi koje treba ubaciti
- Ovaj metod vraća niz sa izbačenim elementima
- Može se koristiti i za uklanjanje elemenata iz niza

Sortiranje nizova

- **sort()** metod sortira niz po abecednom red

```
var niz = ["jabuka", "banana", "trešnja", "kivi", "jagoda"];
niz.sort();
```

- **sort()** metod sortira sve vrednosti kao stringove:
  - ananas ide pre banane
  - prvo ide 100 pa 25, jer je 1 manje od 2
- Da se ispravi ta greška koristimo funkciju poređenja

Sortiranje nizova - funkcija poređenja

- Funkcija poređenja služi da odredi redosled sortiranja
- Ona treba da vraća pozitivan broj, negativan broj ili nulu.

```
function(a, b){return a - b}
```

- Kada **sort()** poredi dve vrednosti, pošalje ih funkciji poređenja. Ukoliko je rezultat negativan, onda stavi a ispred b, ako je pozitivan onda je b ispred a, ako je 0 onda nema promene

Metode za prolazak kroz ceo niz

- `forEach()` metod poziva funkciju jednom za svaki elemenat niza

```
var brojevi = [-40, 100, 1, 5, 0, 10];
brojevi.forEach(mojaFunkcija);

function mojaFunkcija(value) {
  //neki kod koji se izvršava nad svakim elementom niza
}
```

- `map()` metod kreira novi niz izvršavajući funkciju nad svakim elementom niza

```
var brojevi2= brojevi.map(mojaFunkcija);
function mojaFunkcija(value) {
  return value * 2;
}
```

Metode za prolazak kroz ceo niz

- `filter()` kreira novi niz sa elementima koji zadovoljavaju neki uslov

```
var veciOd10 = brojevi.filter(mojaFunkcija);
function mojaFunkcija(value) {
    return value < 10;
}
```

- `reduce()` prolazi kroz ceo niz da izračuna jednu vrednost

```
var zbir = brojevi.reduce(mojaFunkcija);
function mojaFunkcija(total, value) {
  return total + value;
}
```

Hvala na pažnji!