

JavaScript za uporabo na spletnih straneh – osnove

JavaScript trditve

- `var x, y, z;` `// prva trditev`
- `x = 3;` `// druga trditev`
- `y = 6;` `// tretja trditev`
- `z = x + y;` `// četrta trditev`

Program je seznam navodil računalniku, ki naj jih izvede.

V programerskem jeziku tem navodilom pravimo trditve.

JavaScript program je seznam trditev (JavaScript koda).

V HTML-ju JavaScript izvaja brskalnik.

JavaScript trditve

- JavaScript trditve so sestavljene iz:
 - vrednosti,
 - operatorjev,
 - izrazov,
 - ključnih besed in
 - komentarjev.

Primer:

```
Document.getElementById("demo").innerHTML = „Zdravo, Ida.“;
```

Večina JavaScript programov vsebuje veliko trditev. Trditve se izvajajo ena za drugo, v takšnem vrstnem redu, kot so zapisane.

Zapis JavaScript trditvev

- Trditve ločujemo s podpičjem. Podpičje dodamo na koncu vsake trditve!

```
var a, b, c;    // določi 3 spremenljivke (variable)
a = 5;          // pripiše vrednost 5 spremenljivki a
b = 6;          // pripiše vrednost 6 spremenljivki b
c = a + b;      // pripiše vsoto spremenljiv a in b spremenljivki c
```

- Trditve lahko pišemo eno za drugo v isto vrstico, vendar zaradi preglednosti kode to odsvetujem:

```
a = 5; b = 6; c = a + b;
```

- Prav tako niso potrebni presledki v zapisu trditve, vendar so zaradi preglednosti kode zaželeni. Oba naslednja zapisa sta pravilna:

```
var oseba = „Stane“;
var oseba=„Stane“;
```

Zapis JavaScript trditev

- Dolžina vrstice ni pomembna, vendar jo ponavadi omejimo na 80 znakov, zaradi preglednosti kode. Če je trditev predolga, jo je najbolje prekiniti za operatorjem:

```
document.getElementById("demo").innerHTML =  
    „Zdravo, Ida!";
```

- Bloki kode – trditve lahko pišemo skupaj, znotraj zavrtih oklepajev {...}, z namenom, da jih izvedemo skupaj. Tak primer so trditve znotraj funkcije:

```
function mojaFunkcija() {  
    document.getElementById("demo1").innerHTML = „Zdravo, Ida!";  
    document.getElementById("demo2").innerHTML = „Kako si?";  
}
```

Ključne besede v JavaScriptu

- **break:** prekine zamenjavo ali zanko
- **continue:** zapusti zanko in začne na začetku
- **debugger:** prekine izvedbo JavaScript kode in priključ (če je dostopna) funkcijo za iskanje napak
- **do ... while:** izvede blok trditev in ga vedno znova izvaja, dokler pogoj drži (je resničen)
- **for:** označi blok trditev, ki naj se izvajajo, dokler je pogoj izpolnjen
- **function:** deklarira funkcijo
- **if ... else:** označi blok trditev, ki naj se izvajajo, glede na dani pogoj
- **return:** zapusti funkcijo
- **switch:** označi blok trditev, ki naj se izvajajo, odvisno od različnih primerov
- **try ... catch:** implementira upravljanje napake v blok trditev
- **var:** deklarira variablo (spremenljivko)

Ključne besede v JavaScriptu

Ključne besede se uporabljajo za izvedbo posameznih nalog.

Ključne besede so rezervirane besede in jih ni moč uporabljati kot imena spremenljivk (variabel)!

Sintaksa JavaScripta

- Sintaksa so pravila, na kakšen način mora biti program zapisan.

```
var x, y;      // tako deklariramo spremenljivko  
x = 5; y = 6;  // tako določimo vrednost  
z = x + y;     // tako združujemo vrednosti
```

- JavaScript pozna dva tipa vrednosti, fiksne (absolutne) in spremenljive vrednosti.

Fiksnim vrednostim pravimo dobesedne vrednosti (angl. literals) in spremenljivim vrednostim spremenljivke ali variable (angl. variables).

Fiksne vrednosti

- Najbolj pomembna pravila za zapis fiksnih vrednosti so:
 - Števila
 - Nizi
- **Števila** lahko zapisujemo kot cela števila ali kot decimalna števila:
 - 1001
 - 10.50
- **Nizi** so besedilo, ki ga pišemo znotraj narekovajev (enojnih ali dvojnih – JavaScript ne ločuje med enimi ali drugimi):
 - „Neznana oseba“
 - ‚Neznana oseba‘

Spremenljive vrednosti

- Spremenljivke uporabljamo za shranjevanje vrednosti podatkov.
- Spremenljivke deklariramo s ključno besedo **var**:
`var x;`
- Za dodajanje vrednosti spremenljivki, uporabimo znak za enačaj:
`x = 6`

Operatorji

- JavaScript uporablja aritmetične operatorje, kot so +, -, * in /, za računanje vrednosti:

$(5 + 6) * 10$

- Znak = se uporablja za dodajanje vrednosti spremenljivkam, ne za izračun!

```
var x, y;  
x = 5;  
y = 6;
```

Izrazi

- Izraz v JavaScriptu je kombinacija vrednosti, spremenljivk in operatorjev, ki nam vrnejo neko vrednost.
- Primer:
 - $5 * 10$ nam vrne vrednost 50
 - Izrazi lahko vsebujejo tudi spremenljivke, kot npr. $x * 10$, ali nize, kot je npr. „Neznana“ + „ „ + „oseba“, ki nam vrne vrednost „Neznana oseba“.

Komentarji

- Komentarji, zapisani znotraj kode, se ne izvajajo.

- Za zapis komentarja uporabimo znak //.

Vse, kar je zapisano za tem znakom, bo brskalnik razumel kot komentar in kode ne bo izvedel.

// to je komentar

- Več vrstic komentarja zapisujemo med znaka /* in */

/*

Tudi to je
komentar

/*

Identifikatorji

- Identifikatorji so imena.
- Uporabljamo jih, da poimenujemo spremenljivke, ključne besede, funkcije in oznake.
- Imena lahko vsebujejo črke, številke, znak za podčrtaj in znak za dolar, vendar prvi znak imena ne sme biti številka! Na ta način JavaScript hitro prepozna razliko med identifikatorjem in številko.
- JavaScript ločuje med velikimi in malimi črkami, zato, npr. spremenljivki ime in Ime, obravnava kot dve različni spremenljivki.
- Imena ne smejo vsebovati presledka!
- Ključnih besed ne moremo uporabiti za imena!

Spremenljivke

- Spremenljivke so kontejnerji za shranjevanje vrednosti podatkov.
- Primer:

```
var x = 5;  
var y = 6;  
var z = x + y;
```

V tem primeru so x, y in z spremenljivke. X hrani vrednost 5, y hrani vrednost 6 in z hrani vrednost 11.

Kakor v matematiki, imamo tudi tukaj spremenljivke, ki imajo neko vrednost, kot npr. x.

In, kakor v matematiki, tudi tukaj uporabljamo izraze, ki izražajo neko vrednost, kot npr. $z = x + y$.

Spremenljivke

- Znak za enačaj (=) uporabljamo za dodajanje vrednosti spremenljivki, ne za izračun! Za izračun se uporablja dvojni enačaj (==).
- Spremenljivke lahko imajo vrednosti, ki predstavljajo številke, ali besedilne vrednosti. (JavaScript pozna različne vrste podatkov, vendar se bomo za zdaj osredotočili le na ti dve.)
- Besedilnim vrednostim pravimo nizi in jih zapisujemo med enojne ali dvojne narekovaje. Številke ne zapisujemo med narekovaje, ker če jih, jih JavaScript obravnava kot niz in ne številko!

```
var pi = 3.14;
```

```
var oseba = „Neznana oseba“;
```

Spremenljivke

- Spremenljivko deklariramo s ključno besedo **var**, ki ji dodamo unikatno ime, npr.:
var avto;
- Če spremenljivki ne pripišemo vrednosti, nam bo vrnilo rezultat **undefined** (slov. nedefinirano), ko jo bomo priklicali.
- Vrednost spremenljivki pripišemo z znakom za enačaj, npr.:

var avto = „fičo“;

- **Primer:**

```
<p id="demo"></p>
```

```
<script>
```

```
var avto = „fičo“;
```

```
document.getElementById("demo").innerHTML = avto;
```

```
</script>
```

Spremenljivke

- Spremenljivke lahko zapišemo v eni trditvi (v eni vrstici), ali vsako v svoji. Oboje je pravilno, vendar zaradi preglednosti kode priporočam pisanje vsake spremenljivke v svojo vrstico.

- Primer 1:

```
var oseba = „Neznana oseba“, avto = „fičo“, cena = 200;
```

- Primer 2:

```
var oseba = „Neznana oseba“,  
    avto = „fičo“,  
    cena = 200;
```

Spremenljivke

- S pomočjo spremenljivk lahko tudi računamo:

```
var x = 5 + 2 + 3; // rezultat bo 10
```

- Seštevamo lahko tudi besede, kar pomeni, da nam jih bo izpisalo eno za drugo:

```
var x = „Neznana“ + " " + „oseba“; // rezultat bo Neznana oseba
```

- Kadar začnemo niz s številko v narekovaju (pomeni, da jo bo JavaScript obravnaval kot besedo), bo tudi vse naslednje številke obravnaval kot besede:

```
var x = "5" + 2 + 3; //rezultat bo 523
```

- V naslednjem primeru bo najprej seštel številke in potem pripisal številko v narekovaju:

```
var x = 2 + 3 + "5"; //rezultat bo 55
```