

INTERNET TEHNOLOGIJE I VEB DIZAJN

Obrada događaja u JavaScript-u

predavač: dr Miloš Antonijević

Događaji (events) u JavaScript-u

- Događaji u JavaScript-u definišu tip, vreme i mesto obavljanja interakcije korisnika sa Veb stranicom.

onAbort	prekid učitavanje slike
onBlur	izlazak iz fokusa elemenata
onClick	klik mišem
onDbClick	dupli klik mišem
onChange	promena vrednosti u polju
onError	greška pri učitavanju
onFocus	element je dobio fokus
onKeyDown	dugme tastature je pritisnuto
onKeyPress	dugme tastature je pritisnuto ili otpušteno
onKeyUp	dugme tastature je otpušteno

Događaji (events) u JavaScript-u (2)

onLoad	završeno učitavanje Veb stranice
onMouseDown	dugme miša je pritisnuto
onMouseMove	kursor miša je pomeren
onMouseOut	pomeranje miša van elementa
onMouseOver	miš je pomeren preko elementa
onMouseUp	dugme miša je otpušteno
onReset	aktiviranje tastera reset u obrascu
onSelect	tekst polja je selektovan
onSubmit	aktiviranje tastera submit u obrascu
onUnload	korisnik napušta Veb stranicu

Spisak tipova događaja po tagovima

<A>

- click (onClick)
- mouseOver (onMouseOver)
- mouseOut (onMouseOut)

<AREA>

- mouseOver (onMouseOver)
- mouseOut (onMouseOut)

<BODY>

- blur (onBlur)
- error (onError)
- focus (onFocus)
- load (onLoad)
- unload (onUnload)

<FORM>

- submit (onSubmit)
- reset (onReset)

- abort (onAbort)
- error (onError)
- load (onLoad)

<INPUT TYPE = "button">

- click (onClick)

<INPUT TYPE = "checkbox">

- click (onClick)

<INPUT TYPE = "reset">

- click (onClick)

<INPUT TYPE = "submit">

- click (onClick)

<INPUT TYPE = "text">

- blur (onBlur)
- focus (onFocus)
- change (onChange)
- select (onSelect)

<SELECT>

- blur (onBlur)
- focus (onFocus)
- change (onChange)

<TEXTAREA>

- blur (onBlur)
- focus (onFocus)
- change (onChange)
- select (onSelect)

Primer prikaza okinutih događaja

```
<div id="test_div" onclick="prikazi_dogadjaj(event, 'onclick')" onmouseover="prikazi_dogadjaj(event, 'onmouseover')">Isprobavanje dogadjaja</div>
```

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

```
<script>
```

```
function prikazi_dogadjaj(e, tekst) {
```

```
    //document.getElementById("display_area").innerHTML=tekst;
```

```
    document.getElementById("display_area").innerHTML=event.type;
```

```
}
```

```
</script>
```

Primer pribavljanja položaja miša

```
<div id="test_div" onmousemove="prikazi_poziciju(event)">Isprobavanje  
dogadjaja</div>
```

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

```
<script>
```

```
function prikazi_poziciju(e) {
```

```
    var x = e.clientX;
```

```
    var y = e.clientY;
```

```
    var cursor = "Pozicija misa je : " + x + " and " + y ;
```

```
    document.getElementById("display_area").innerHTML=cursor
```

```
}
```

```
</script>
```

Primer onBlur događaja

```
<form name="forma">
    <input type="text" name="broj" value="" onBlur="valid()">
</form>

<script>
function is_num_valid(){
    var input = document.forma.broj.value;
    if (input < 0) alert("Uneli ste negativan ceo broj");
}
</script>
```

Primer onChange događaja

```
<input type="text" onfocus="this.oldvalue = this.value;"  
onChange="onChangeTest(this); this.oldvalue = this.value;" />
```

```
<script>
```

```
function onChangeTest(textbox) {
```

```
    alert("Nova vrednost je: " + textbox.value + "\n" + "Stara vrednost je  
bila: " + textbox.oldvalue);
```

```
}
```

```
</script>
```


Primer onClick događaja

```
<form name="forma">  
    <input type="text" name="data" size=10 />  
    <input type="button" value="KLIKNI TE" onClick="prikazi_pozdrav()">  
</form>
```

```
<script>  
    function prikazi_pozdrav() {  
        var input=document.myform.data.value;  
        alert("Pozdrav " + input + " ! ");  
    }  
</script>
```

Primer onMouseOver, onMouseOut i KeyDown, KeyUp događaja

```
<br>
<a href="primer.html" onMouseOver="document.images[0].src='1.png'"
onMouseOut="document.images[0].src='2.png';" >Link koji menja sliku</a>

<input type="text" onkeyup="prikazi_dogadjaj(event)"
onkeydown="prikazi_dogadjaj(event)">
<p id="display_area"></p>
<script>
    function prikazi_dogadjaj(e, tekst) {
        document.getElementById("display_area").innerHTML=event.type;
    }
</script>
```

Dinamičko registrovanje događaja

- Događaji u JavaScript-u se mogu dinamički pridružiti nekom elementu korišćenjem **addEventListener()** metode elementa.
- Prvi argument metode predstavlja naziv događaja koji se registruje, a drugi argument naziv funkcije koja se poziva kada se događaj okine (trigger).

```
<script>
    function getEventType(event) {
        console.log(event.type);
    }
</script>
document.addEventListener('keydown', getEventType);
document.addEventListener('keypress', getEventType);
document.addEventListener('keyup', getEventType);
```

Rad sa String-ovima

- Kada se u promenjivu smeštaju podaci prikupljeni iz input elementa, primenjiva dobija tip String.
- Ako je potrebno tretirati podatke kao broj, podaci se moraju konvertovati iz String-a u float (sa decimalnom tačkom) ili integer (ceo broj), korišćenjem **parseInt()**, odnosno **parseFloat()** metoda.

```
var prva_vrednost = document.forma.broj1.value;  
var druga_vrednost = document.forma.broj2.value;
```

```
console.log(prva_vrednost + druga_vrednost);  
console.log(parseInt(prva_vrednost) + parseInt(druga_vrednost));
```

Metode za rad sa karakterima String-a

- Prebrojavanje broja karaktera se obavlja korišćenjem **length** svojstva promenjive String (String je **niz** karaktera).
- Pribavljanje karaktera koji se nalazi na određenoj poziciji se obavlja upotrebom metode **charAt()**. Argument metode predstavlja indeks traženog karaktera.
- Pribavljanje grupe karaktera koji se nalaze od pozicije x do pozicije y se obavlja korišćenjem **substring()** metode. Prvi argument metode je indeks prvog karaktera (included), a drugi argument je indeks drugog karaktera (excluded). Ako se navede samo jedan argument, pribavlja se niz karaktera počev od navedenog indeksa do kraja String-a.
- Dohvatanje grupe karaktera dužine y, počev od pozicije x se obavlja upotrebom metode **substr()**. Prvi argument predstavlja indeks početnog karaktera, a drugi argument predstavlja dužinu. Ako je prvi argument negativan broj, brojanje počinje od kraja String-a. Ako se navede samo jedan argument, pribavlja se niz karaktera počev od navedenog indeksa do kraja String-a.

Rad sa String-ovima (2)

- Traženje prve pozicije nekog stringa u okviru drugog stringa se obavlja korišćenjem metode **indexOf()**.
- Metoda `indexOf()` može da prihvata jedan ili dva argumenta. Prvi argument metode predstavlja traženi string. Ako se navede samo jedan argument, pretraga se obavlja od početka stringa. Ako se navede drugi argument (ceo broj), pretraga se započinje od navedenog indeksa karaktera.

```
var neki_string = "Mika ima kamion!";
```

```
console.log(neki_string.indexOf("ka"));
```

```
console.log(neki_string.indexOf("ka", 4));
```

Matematičke funkcije

- Matematičkim funkcijama u JavaScript-u se pristupa putem klase **Math**.
- Generisanje slučajnog broja je može obaviti korišćenjem funkcije **random()** koja vraća decimalni broj između 0 i 1.
- Funkcije **min()** i **max()** vraćaju najmanji, odnosno najveći broj od brojeva navedenih u listi argumenata.
- Funkcije **round()**, **ceil()** i **floor()** obavljaju zaokruživanje decimalnih brojeva. I to:
 - **round()** vrši zaokruživanje na najbliži ceo broj (1.4 => 1, 1.6 => 2)
 - **ceil()** vrši zaokruživanje uvek na prvi viši broj (1.4 => 2, 1.6 => 2)
 - **floor()** vrši zaokruživanje uvek na prvi niži broj (1.4 => 1, 1.6 => 1)

Odloženo izvršavanje u JS

- Odloženo izvršavanje komandi na osnovu vremenskog brojača (timer) se može izvesti korišćenjem metoda **setTimeout()** i **setInterval()** objekta window.
- setTimeout() metoda sadrži dva atributa. Prvi atribut određuje koji će se izraz izvršiti jednom nakon isteka broja milisekundi naznačenih u drugom argumentu.

```
<input type="button" value="Klikni me!"  
onclick="setTimeout('alert(\'Surprise!\')', 3000)"/>
```

- setInterval() metoda izvršava izraz naveden u prvom argumentu na svakih X milisekundi koje su navede u drugom argumentu metode.

```
<p id="brojac">0</p>  
<button onclick="myFunction()">Try me</button>  
<script>  
function myFunction() {  
    setInterval(function(){ document.getElementById("brojac").innerHTML =  
parseInt(document.getElementById("brojac").innerHTML) + 1 }, 1000);  
}  
</script>
```


Eval metoda

- Metoda **eval()** (objekta window) izvršava vrednost izraza navedenog u argumentu kao da je u pitanju JavaScript kod.

```
var a = 5;  
document.write(eval("1+2"));  
document.write("<br>");  
eval("c=1; d=3; document.write(c+d)");  
document.write("<br>");  
document.write(eval(a+2));
```

PITANJA?