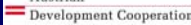




right to education

COURSE DEVELOPMENT PROGRAM +

financed by



Programiranje za Internet



predavanja
v.as.dr. Samir Lemeš
slemes@mf.unze.ba

23. JavaScript

- Funkcije
- Objekti
- Obrada događaja

Funkcije

- Za kreiranje vlastite funkcije, koristi se naredba **function**.
- Naprimjer, sljedeća funkcija ispisat će tekst koji primi kao parametar bold slovima:


```
function ispisuj_bold (tekst) {
    document.write ("<B>" + tekst + "</B>");
}
```
- U JavaScriptu se može formatirati ekranski ispis koristeći HTML naredbe, kao u gornjem primjeru.
- Skripta će browseru poslati cijeli string, koji uključuje i HTML naredbe za određivanje stila ispisa, a browser će to ispisati po njemu poznatim pravilima, koja kažu da sve između i mora biti bold.

Funkcije

- Novu funkciju pozvat ćete navođenjem imena i parametra:
Ispisuj_masno ("Želim ocjenu 10!");
- Ova funkcija može ispisivati i cijele brojeve, i brojeve s pomičnim zarezom te stringove i sve ostalo.
- Ovo je posljedica toga da JavaScript automatski pretvara podatke među različitim tipovima.
- **Ako funkcija prima više parametara, prilikom njenog pozivanja morate joj poslati onoliko njih koliko ih očekuje!**

Stvaranje objekata

- Prvo je potrebno napisati funkciju:


```
function knjiga (autor, izdavac, stranica, CD, sifra) {
    this.autor=autor;
    this.izdavac=izdavac;
    this.stranica=stranica;
    this.CD=CD;
    this.sifra=sifra;
}
```
- koja će pojedinim svojstvima objekta pridružiti odgovarajuće vrijednosti. Oznaka "this" samo naznačava da se mijenjaju svojstva trenutnog objekta, a ne nekog drugog.
- Nakon toga možete stvoriti novi objekt tijekom izvršavanja programa pomoću operatora new:


```
novaknjiga=new knjiga ("XY", "UNZE", 1050, true, 01);
```

Ugrađeni objekti

- Objekt **Window** sadrži funkcije pomoću kojih možete utjecati na izgled prozora browsera u kojem se izvršava vaš JavaScript program.
- Primjer:


```
window.frames
['okvir1'].document.bgColor="yellow"
```

 mijenja boju pozadine u trenutnom dokumentu, ali samo u njegovom okviru **okvir1**.
- Prvi dio te naredbe, **window.frames**, koristi se za pristup nekom okviru, a nakon toga treba pozvati funkciju kojom se postavlja nova boja:


```
"okvir1" oboji u žuto
```

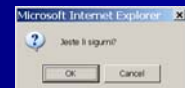
Ugrađeni objekti

- Da biste prikazali poruku u statusnoj traci pretraživača, možete pozvati
`window.status="Ne mogu zaspati bez neke TV serije!";`
- Ako želite prikazati okvir za dijalog s nekim upozorenjem, jednostavno pozovite
`window.alert("Ažuriraj antivirusni program!");`



Ugrađeni objekti

- Ako želite prikazati okvir za dijalog u kojem korisnik može birati između potvrde i odbijanja pozovite funkciju
`window.confirm("Jeste li sigurni?");`
- OK programu vraća vrijednost true, a Cancel vrijednost false.
- Da biste otvorili novi prozor i u njega učitali neki dokument, pozovite funkciju `open`:
`open("stranica.htm","Window1","toolbar=yes");`



Ugrađeni objekti

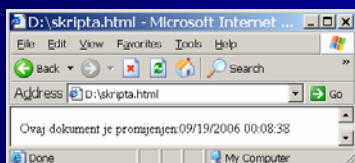
- Objekt **Document** daje osnovne informacije o trenutnom dokumentu i koristi se za sve ulazno-izlazne operacije.
- Da biste ispisali neki tekst, upotrijebite naredbe `document.write` i `document.writeln`:
`document.write("Moja prva skripta");`
`document.writeln("Moja prva skripta");`
- Razlika između prve i druge naredbe je u tome što će se u drugom slučaju tačka od koje počinje ispis nekog sljedećeg stringa premjestiti u novi red.

Ugrađeni objekti

- Da biste promijenili boje kojima se označavaju hipertekstualne veze (uključujući i one koje su već posjećene), upotrijebite:
`document.linkColor='red';`
`document.alinkColor='yellow';`
`document.vlinkColor='blue';`
- Koristeći JavaScript možete saznati i odakle je došao vaš posjetilac, tj. s koje je stranice odabrao hiperlink koji je pokazivao na vašu stranicu:
`document.write (document.referrer);`

Ugrađeni objekti

- Ako želite ispisati datum zadnje promjene trenutnog dokumenta:
`document.write ("Ovaj dokument je promijenjen:");`
`document.write (document.lastModified);`



Ugrađeni objekti

- Objekti **Date** i **Math** sadrže uobičajene funkcije za rukovanje datumima i vremenom, te obavljanje matematičkih operacija.
- U objektu **Math** postoje sve funkcije kao i u Javinoj klasi **Math** (`sin`, `cos`, `tan`, `asin`, `acos`, `atan`, `exp`, `log`, `sqrt`, `pow`, `round`, `random`, `abs`, `max`, `min`). Funkcije se pozivaju na uobičajeni način:
`rezultat = Math. Cos (2.3564);`
- Novi objekt koji sadrži datum stvorit ćete s:
`datum = new Date (parametri);`

Ugrađeni objekti

- Ako parametri nisu navedeni, objekt će sadržavati trenutni datum i vrijeme. Datum možete navesti u obliku stringa u obliku `mjesec dan, godina sati:minute:sekunde`
`"12. 1, 1974 22:00:00"`
- Treća je mogućnost da navedete cijele brojeve koji redom predstavljaju godinu, mjesec, dan, sate, minute i sekunde, i to tačno tim redoslijedom. Kod navođenja mjeseca i dana u JavaScriptu oni se numerišu od 0, a ne od 1!

Obrada događaja

- Događaji su akcije koje obavlja korisnik.
- Korištenje JavaScripta za provjeru ispravnosti popunjavanja obrazaca:
- U HTML naredbu treba uključiti poziv tog događaja:
`<INPUT TYPE="button" VALUE="posalji" onClick="posalji(this.form)">`
- Novi parametar u naredbi `<INPUT>` govori koji događaj treba pratiti.

Obrada događaja

blur	onBlur	...kada korisnik premjesti fokus s trenutnog elementa na Web stranici
click	onClick	...kada korisnik klikne na neki element ili hiperlink
change	onChange	...kada korisnik promijeni neki tekst ili mišem označi neki element na Web stranici
focus	onFocus	...kada se korisnik usredotoči na neki element (npr. kada u obrascu za upis podataka klikne unutar nekog od polja)
load	onLoad	...kada korisnik ušita stranicu u browser
mouseover	onMouseOver	...kada korisnik mišem dođe na hiperlink
select	onSelect	...kada korisnik označi neke elemente u obrascu
submit	onSubmit	...kada korisnik pošalje podatke upisane u obrazac
unload	onUnload	...kada korisnik napušta stranicu

Obrada događaja

- Umjesto da funkciju pozivate, ako se radi o vrlo kratkoj naredbi koju treba izvršiti nakon određenog događaja, možete je izravno uključiti u akciju koju treba obaviti nakon događaja.
- Naprimjer, da biste pokazali okvir s upozorenjem ako korisnik klikne na neku tipku:
`<INPUT TYPE="button" name="Klikni!" onClick="alert('Zašto bezveze klikaš?')">`
- Ili biste mogli na svojoj stranici napraviti dugme koji će pretraživaču narediti da se vrati dvije stranice nazad, ovisno o tome gdje je korisnik bio:
`<INPUT TYPE="button" name="Klikni!" onClick="history.go(-2)">`

Obrada događaja

- Ako želite pozvati neku funkciju nakon što korisnik odluči poslati sadržaj popunjenog obrasca, upotrijebite događaj `onSubmit`:
`<INPUT NAME="obrazac" ACTION="obrada.cgi" onSubmit="provjeri ( )">`
- Ovaj primjer će pozvati funkciju `provjeri( )` u vašem programu, pa tako možete provjeriti je li korisnik ispravno popunio sva polja.

Obrada događaja

- Kada korisnik učini neku promjenu u obrascu (npr. odabere neku drugu opciju između ponuđenih ili promijeni sadržaj nekog polja za upis teksta), JavaScript poziva `onChange`:
`<INPUT NAME="adresa" onChange="provjeriAdresu ( )">`
- Ovakva naredba pozvala bi funkciju `provjeriAdresu` svaki put nakon što korisnik promijeni "fokus" odnosno nakon što je upisao nešto u neko polje i premjestio se na drugo.
- Na ovaj način možete provjeriti je li dobro napisao svoje lične podatke (npr. telefonski broj ne može imati 3 cifre, adresa mora sadržavati barem dva broja, broj ulice i poštanski broj grada, itd).
- Korisnika možete stalno vraćati na isto polje za upis sve dok ne ispravi pogrešku.

Obrada događaja

- Ako želite da se neka poruka ispiše čim korisnik dođe na vašu Web stranicu, upotrijebite događaj **onLoad**.
- Njega je najbolje uključiti u naredbu <BODY> u HTML dokumentu.
- Sljedeći primjer otvorio bi okvir za dijalog i u njemu prikazao poruku odmah nakon što korisnik učita stranicu koja sadrži taj redak:
`<BODY onLoad=alert('Dajemo popust za sve narudžbe na Webu!')>`
- A ako želite korisnika pozdraviti na odlasku ili obaviti neku drugu radnju, možete to na sličan način učiniti pomoću događaja **onUnload**.
`<BODY onLoad=alert('Hvala na posjeti!')>`

Obrada događaja

- Događaj **onMouseOver** radi slično prethodnima: nakon što se mišem nađete iznad određenog objekta, JavaScript će obaviti neku radnju. Na primjer:
`<INPUT TYPE="button" name="Klikni!"
onMouseOver="help()">`
- Na ovaj način biste mogli ispisivati kratke poruke pomoći koje korisniku objašnjavaju što pojedini elementi stranice rade.