



Internet programiranje

Čas 3 - HTML forme, frejmovi
i layout

Dražen Drašković,
drazen.draskovic@etf.bg.ac.rs

HTML forme, frejmovi i layout

FORME

Uvod u forme

- Forme nam služe za prenos podataka sa klijentske strane ka serveru. Dobijeni podaci se mogu prihvatiti, zapamtiti ili editovati.
- Tehnologije koje se mogu primeniti na serverskoj strani aplikacije su:
 - **ASP .NET** (Active Server Pages),
 - **PHP** (*PHP*: Hypertext Preprocessor),
 - **JSP** (Java Server Pages) i servleti,
 - ...

Uvod u forme

- Podaci se na klijentskoj strani unose pomoću koncepta formi (eng. *form*).
- Forma dopušta da se uspostavi komunikacija između korisnika i servera:
korisnik popunjava formu i šalje je ka serveru.
- Ona omogućava da se u okviru servera pozove određeni programski modul, koji će nakon osnovne obrade podataka generisati najčešće novu HTML klijentsku stranicu.

Tag <form>

- Osnovni tag <**FORM**>
- Opšti oblik ovog taga je:

<FORM>

<!-- definicija elemenata forme -->

</FORM>

- Atribut *name* dodeljuje jedinstveno ime formi
- Atribut *action* treba da označava URL adresu na serveru na koju šaljemo podatke

Atribut method

- **METHOD** predstavlja metodu prenosa argumenata programa.
- Ima vrednosti *GET* ili *POST*
- Korišćenjem **GET** vrednosti parametara koje se šalju ka serveru navode se u okviru adrese programa koji se poziva.
Na primer
<http://server.com/ServerskiProgramskiModul?parametar1=88>
gde se šalje vrednost promenljive *parametar1* kao 88.
- Metoda **GET** ima ograničen broj parametara: mora važiti da je *dužina URL + dužina parametara < 1KB*.
- Metod **POST** vrednosti elemenata forme šalje u okviru zaglavlja poruke koju šalje na server, tako da nisu vidljivi korisniku i nema ograničenja u količini podataka.
- Programski modul na serveru trebalo bi da zna kojim metodom prima podatke od strane klijenta, GET ili POST.
- U praksi se češće koristi metod **POST**, posebno za podatke koji ne treba da budu javni, kao što su korisničko ime i šifra, matični broj, broj kreditne kartice,...

Elementi forme

- U okviru same forme definišu se različiti elementi forme: tekstualna polja za unos podataka, tekst prostore za komentare, polja za potvrdu, padajuće liste, različite vrste dugmadi,...
- Treba napomenuti da se **jedino u okviru jedne forme ne može definisati druga forma!!!**

Atributi elemenata forme

- Potrebno je da svaki element forme, pored *tipa (type)*, ima i definisane dve karakteristike:
 - ime elementa forme (*name*),
 - vrednost elementa (*value*).
- Ove dve karakteristike su zajednički za skoro sve tagove koji predstavljaju elemente forme.
- *Name* treba da bude jedinstven za svaki element, da bismo mogli da pročitamo ili izmenimo vrednost tog elementa.

U okviru forme moguće je definisati:

- Tekstualna polja
- Komandna dugmad,
dugme za potvrdu ili poništavanje forme
- Tekstualne prostore
- Polja za potvrdu
- Radio dugmad
- Padajuće liste
- Skrivenne promenljive

(Primer 1)

Tekstualno polje

- Služi za unos tekstualnih informacija koje se kasnije šalju na server, kao što su ime, prezime, adresa mesta stanovanja,...
- Piše se pomoću taga <INPUT>, s tim što se koristi atribut *type* sa definisanom vrednošću **text**.
- Osnovni oblik ovog elementa je:

```
<INPUT TYPE="text" NAME="ime"  
VALUE="Unesite ime i prezime" SIZE="30">
```

Atributi za tekst polje

- ***value*** - definiše početni tekst koji će se pojaviti u tekstualnom polju.
- ***size*** - zadaje broj karaktera koji predstavlja širinu polja za unos teksta.
- ***name*** - definiše ime elementa pomoću koga drugi elementi aplikacije (*JavaScript* ili serverski deo aplikacije) mogu čitati ili menjati unetu vrednost u okviru tekst polja.
- ***maxlength*** - definiše maksimalan broj karaktera koji se mogu uneti u tekstualno polje.
- Treba naglasiti da su sva četiri navedena atributa opcionalna i da se mogu i izostaviti.

Unos lozinki u tekstualno polje

- Ako se želi da se zamaskira tekst koji korisnik unosi u tekstualno polje **umesto vrednosti *text*, treba koristiti vrednost *password*.**
- Izgled polja će biti isti kao kod običnog tekst polja, ali će svaki uneti znak biti prikazan u obliku *.
- Sam čitač pamti unete karaktere i ostatku aplikacije se predaje tekst koji je korisnik uneo, a ne niz simbola *.

Komandno dugme

- Postoje više vrste dugmadi za različitim funkcijama.
- Sve vrste imaju zajednički tag **<INPUT>**, u okviru koga se pomoću atributa **type** definiše funkcija.

Najčešće korišćene vrednosti atributa *type* su:

- **button** - obično komandno dugme,
- **submit** - dugme za potvrdu forme,
- **reset** - dugme za poništavanje forme.

<input type="button">

- Definiše komandno dugme opšteg tipa.
- Pritiskom na ovo dugme korisnik izvršava akciju koja se mora definisati u okviru samog taga.
- Najčešće je to JavaScript funkcija koja izvršava određene provere pre samog slanja podataka na server:

```
<input type="button" name="Proveri"  
value="OK" onClick="funkcijaZaProveru()">
```

- Kada korisnik klikne na OK, izvršava se Java Script funkcija funkcijaZaProveru(), koja mora biti definisana u okviru iste HTML stranice.

<input type="submit">

- Definiše dugme za potvrdu forme, kojim se podaci šalju na server.
- Podaci se šalju pomoću akcije (atribut ACTION) koja je definisana u okviru taga <FORM>, pa nije potrebno definisati funkciju koja će se pozvati klikom na dugme *Submit*.
- Primer:

```
<input type="submit" value="Potvrdi">
```

<input type="reset">

- Definiše dugme za poništavanje forme, odnosno vraćanje forme u prvobitno stanje.
- I u slučaju *Reset* nije potrebno definisati novu JavaScript funkciju.
- Primer:

```
<input type="reset" value="Ponisti">
```


Radio dugme

- Predstavlja kružić koji korisnik pomoću miša ili tastature može čekirati ili odčekirati.
- Sintaksa elementa:

`<INPUT type="radio">`



- Koristi se kod izbora SAMO jedne opcije od više opcija, na primer: pol (muški ili ženski), navijam za... (Crvenu Zvezdu ili Partizan),...

Primer sa radio

```
<I>Za koji klub navijate?</I><br>  
  <INPUT type="radio" name="tim" value="cz">  
  Crvena Zvezda<br>  
  <INPUT type="radio" name="tim" value="par">  
  Partizan
```

- Obratite pažnju da kod atributa *name*, za istu grupu radio dugmadi, mora da bude isti naziv.
- Time omogućavamo da korisnik odabere samo jednu ili ni jednu opciju (ako ne navija ni za jedan klub).
- Atribut *name* je jako bitan, jer u slučaju da smo ga različito imenovali u ovom primeru, korisnik bi mogao da odabere obe opcije istovremeno, što nije moguće.

Polje za potvrdu - checkbox

- Predstavlja kvadratić (kućicu) koji korisnik pomoću miša ili tastature može štiklirati ili odštiklirati.
- Sintaksa elementa:

`<INPUT type="checkbox">`



Primer sa checkbox-ovima

<I>Koje programske jezike poznajete?</I>

<UL type="square">

 <INPUT type="checkbox" name="cpp"> C++

 <INPUT type="checkbox" name="java"> Java

 <INPUT type="checkbox" name="csharp"> C#

- Nijedan, jedan ili više polja mogu u istom trenutku biti potvrđeni.
- Atribut *name* je jako bitan i koristi se zbog pristupa ostalih delova aplikacije ovom elementu, odnosno proveru da li je polje za potvrdu u formi štiklirano ili ne.
- Ako je *name* isti za sve checkbox-ove, mora da postoji atribut *value* koji je različit.

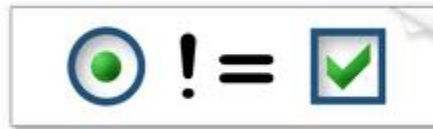
Podrazumevana vrednost

- Radio dugme i checkbox mogu da imaju podrazumevanu (default) vrednost, tako što se dopiše atribut *checked*:

```
<INPUT type="radio" name="pol" value="m" checked> Muski
```

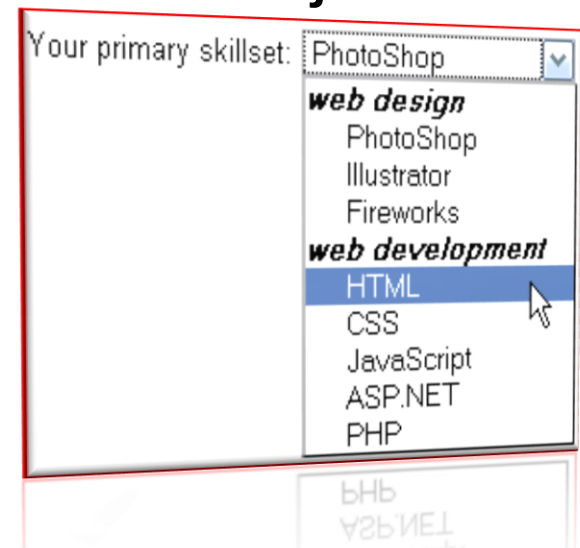
```
<INPUT type="checkbox" name="auto" value="opel" checked> Opel
```

- I zapamtite:



Padajuća lista

- *Drop-down list*, omogućava izbor jedne od više opcija ili izbor više od više opcija (ako je dodat atribut *multiple*).
- U jednom trenutku mu je prikazana samo jedna izabrana opcija, ali klikom miša na strelicu koja je u okviru ovog elementa, korisnik dobija listu svih mogućnosti.
- Koriste se tagovi **<SELECT>** za definisanje liste i **<OPTION>** za svaku opciju (stavku) unutar liste.



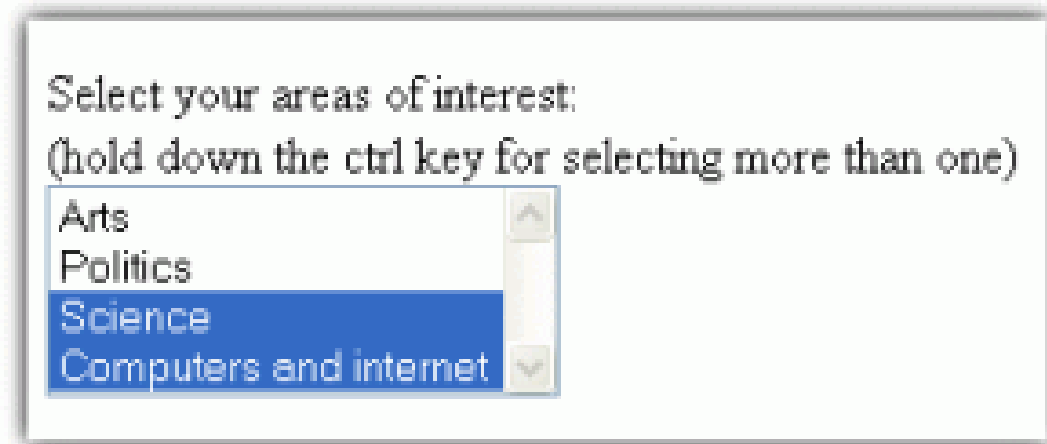
Primer padajuće liste

- Tag **<OPTION>** sa atributom **SELECTED** se uzima kao izabrana vrednost.

```
<B>Moj omiljeni predmet na 1. godini je:</B>  
<select name="predmeti">  
  <option>Analiza</option>  
  <option>Algebra</option>  
  <option selected>Programiranje</option>  
</select>
```

Atributi uz tag <SELECT> su:

- **size** je atribut koji opisuje broj opcija liste, koje će biti prikazane
- **multiple** je atribut koji nema vrednost i kojim se omogućava izbor više opcija u listi (drži se CTRL prilikom izbora opcija)



Tekstualni prostor

- Daje mogućnost da korisnik upiše više linija svog teksta, najčešće svoj komentar.
- Tag **<TEXTAREA>** prikazuje tekst u odvojenom tekst prozoru.
- Dimenzije prozora su određene atributima:
 - **rows** (broj redova teksta) i
 - **cols** (broj kolona).
- Atribut **name** dodeljuje simboličko ime elementa u kome se nalazi tekst.

```
<TEXTAREA name="komentar" rows="6" cols="30">  
Unesite neki komentar </TEXTAREA>
```

Skrivena promenljiva

- Vrednosti koje ne bi trebalo prikazivati korisniku.
- Tag `<INPUT>` i atribut *type* sa vrednošću *hidden* :

```
<input type="hidden" name="serial"  
value="08100-38069">
```

- Ovako definisanoj promenljivoj ostatak aplikacije prilazi pomoću imena, kao i bilo kom drugom elementu forme.

Tag <label>

- Ovaj tag definiše labelu za neki <input> element.
- Vezuje se za atribut **ID** nekog elementa.
- Primer:

```
<label for="element_id">DA</label>  
<input type="radio" name="question"  
id="element_id" value="yes">
```

(Primer 2)

Da li nam znači <label> ?

- Ovaj tag obezbeđuje poboljšanje sa korisničke strane, jer kada definišemo labelu, korisnik može kliknuti mišem i na tekst labele, a ne samo na određeni radio ili checkbox

Pol: Muski ☒ | Zenski ☐



Tag <fieldset>

- Koristi se za grupisanje srodnih elemenata u formi.
- Oko povezanih elemenata iscrtava se kontura (ivica), koja grafički grupiše elemente forme.
- Koristi tag <legend> za naslov grupe elemenata.
- Može da ima atribut *disabled* sa vrednošću *"disabled"* (ovo važi i za ostala polja forme, tada polja nisu editabilna)

(Primer 2)

The screenshot shows a web form titled "Moja druga forma". Inside the form, there is a section titled "Licne informacije:" which is enclosed in a gray border, indicating it is a disabled fieldset. The fields within this section are: "Ime:" with a text input field, "E-mail:" with a text input field, "Datum rođenja:" with a text input field, and "Pol: Muski" with a selected radio button and "Zenski" with an unselected radio button. Below these fields is a "Potvrdi" button. The entire form is set against a light gray background.

HTML forme, frejmovi i layout

FREJMOVI

<iframe>

- Frejmovi se koriste da učitaju jednu veb stranicu u okviru druge veb stranice.
- Sintaksa za dodavanje iframe elementa (gde je URL adresa neke odvojene stranice):

```
<iframe src="URL"> </iframe>
```

Osnovni atributi

- **SRC** je adresa HTML dokumenta koji će biti prikazan u zoni tog taga iframe (ugrađen u frejm)
- **WIDTH** predstavlja širinu elementa iframe
- **HEIGHT** predstavlja visinu elementa iframe
- **FRAMEBORDER** služi za promenu dimenzija ivice elementa iframe (može imati vrednost 0 ili 1)
- **NAME** specificira ime frejma koji se koristi

Upotreba frejmova

- Kada na svojoj stranici želite da imate sadržaj neke druge stranice, tada treba upotrebljavati frejmove.
- Na primer:

```
<IFRAME src="Primer.html" name="MenjaSe">
```

- Ako se želi napraviti link koji će sadržaj prethodno definisanog frejma promeniti i u njega učitati stranicu www.google.com tada treba definisati sledeći kod:

```
<a href="http://www.google.com" target="MenjaSe">  
Promena frejma </a>
```

```
<IFRAME src="Primer.html" name="MenjaSe">
```

- Dakle, frejm mora imati ime, ukoliko želite da menjate njegov sadržaj!

HTML forme, frejmovi i layout

BLOKOVI I LAYOUT

Grupisanje elemenata

- HTML elemente možemo da grupišemo u blokove korišćenjem:
 - `<div>`
 - `<span>`
- Blokovi počinju i završavaju se novim redom, što smo već videli kod mnogih elemenata `<h1>`, `<p>`, `<ul>`,...
- Elementi u jednoj liniji, ne završavaju se novim redom: `<b>`, `<a>`, `<img>`,...

<div> tag

- Tag **<div>** može da se koristi za grupisanje drugih HTML elemenata.
- Zamenjuje stari “način” definisanja izgleda stranice (layout-a) pomoću tabela.
- Svrha taga **<table>** je da prikaže tabelarne podatke!
- Kada se koristi zajedno sa CSS, ovaj tag može da definiše veliki broj atributa čime može uticati na stil (izgled) sadržaja bloka i samog bloka.

(Primer 4) (Primer 5)

** tag**

- Može da se koristi za grupisanje teksta sa određenim stilom.
- Kada se koristi zajedno sa CSS, ovaj tag može da postavi attribute stila na određene delove teksta.

(Primer 6)

Primer layout-a

Internet programiranje

Meni

HTML

CSS

JavaScript

Ovde je neki sadrzaj

Pregled - Vežbe #3

- `<form>`
- `<input type="*">`
 - * zamenjuju: button, submit, reset, text, password, radio, checkbox, hidden
- `<select>` `<option>`
- `<textarea>`
- `<label>`
- `<fieldset>` `<legend>`
- `<iframe>`
- `<div>`
- `<span>`