

Capitolul 11

Formulare

Cu siguranță ați văzut cum arată un formular într-o pagină Web. De exemplu, atunci când completați o carte de oaspeți (*guestbook*) aveți de-a face cu un formular. Atunci când selectați mai multe opțiuni dintr-o listă sau introduceți un cuvânt într-un motor de căutare, folosiți, de asemenea, un formular.

Formularul reprezintă unul dintre cele puternice elemente ale unei pagini Web. Prin intermediul formularelor se realizează interactivitatea cu vizitatorul paginii, acestea permițându-vă să obțineți informații despre cei care vizitează paginile dumneavoastră.

1. Ce este un formular?

Un formular este un ansamblu de zone active alcătuit din mai multe tipuri de elemente: butoane, casete de selecție, câmpuri de editare, etc., ce permit utilizatorilor să introducă efectiv informații. Aceste informații sunt ulterior transmise serverului pe care este găzduită pagina dumneavoastră, unde vor putea fi prelucrate.

O sesiune cu o pagina Web ce conține un formular cuprinde două etape:

- Utilizatorul completează formularul și, prin apăsarea butonului de expediere, trimite serverului datele înscrise în formular.
- O aplicație dedicată de pe server (un script) analizează informațiile transmise și, în funcție de configurația scriptului, fie stochează datele într-o bază de date, fie le transmite la o adresă de mail indicată de dumneavoastră. Dacă este necesar, serverul poate expedia și un mesaj de răspuns utilizatorului.

Cel mai important lucru pe care trebuie să-l înțelegeți în legătură cu formularele este că aici apar două probleme distincte și care necesită instrumente diferite pentru a le gestiona.

Prima dintre ele este plasarea formularului în pagină și asigurarea bunei lui funcționări. Acest lucru se realizează prin HTML și de el ne ocupăm în acest capitol.

Cea de-a doua este gestionarea și prelucrarea informațiilor pe care vizitatorul le introduce prin intermediul formularului. Această problemă este rezolvată cu ajutorul scripturilor CGI stocate pe server, subiect despre care vom vorbi în capitolul următor.

Structura formularelor poate varia, de la o simplă casetă de text pentru introducerea unui șir de caractere, până la un ansamblu complex, cu multiple secțiuni și care oferă facilități puternice de transmitere/prelucrare a datelor.

2. Crearea unui formular

Un formular este definit într-un bloc delimitat de etichetele `<FORM>` `</FORM>`. Eticheta `</FORM>`, de închidere, este obligatorie. În interiorul blocului sunt incluse:

- elementele formularului, în care vizitatorul urmează să introducă informații

- un buton de expediere (*submit*), la apăsarea căruia, datele sunt transmise către server
- opțional, un buton de anulare (*reset, cancel*), prin care utilizatorul poate anula datele înscrise în formular

Tot ceea ce este inclus între eticheta de deschidere și cea de închidere face parte din formular.

Eticheta <FORM> are două atribute esențiale:

action – comunică browserului unde să trimită datele introduse în formular. În general valoarea atributului action este adresa URL a scriptului aflat pe serverul care primește datele formularului. De exemplu:

```
<FORM action="http://www.yahoo.com/cgi-bin/nume_fis.cgi">
```

method - precizează metoda utilizată de browser pentru expedierea datelor formularului.

Atributul method poate avea două valori:

- post - folosită în cele mai multe cazuri. Indică serverului să furnizeze datele direct scriptului ca date de intrare standard.
- get (valoarea implicită) - datele din formular sunt adăugate la adresa URL precizată de atributul action, între adresa URL și date fiind inserat un "?". Datele sunt adăugate conform sintaxei:

```
nume_camp=valoare_camp.
```

Între diferite seturi de date este introdus un "&". De exemplu:

```
"http://www.yahoo.com/cgi-bin/nume_fis.cgi?
nume1=valoare1&nume2=valoare2"
```

Cea mai facilă cale prin care informațiile introduse într-un formular pot parveni creatorului paginii este folosirea comenzii **mailto**: Am mai întâlnit această comandă la capitolul **Legături** în contextul următor:

```
<A href="mailto:autor@domeniu.com">
```

De data aceasta nu mai este vorba de crearea unei legături care să lanseze în execuție aplicația de mail a utilizatorului, ci de transmiterea datelor introduse în formular la o adresă de mail specificată de dumneavoastră.

Acest lucru se poate realiza precizând ca valoare a atributului action următoarea secvență: "mailto:adresa_mail", ca în exemplul următor:

```
<FORM action=mailto:autor@domeniu.com method="post">
```

Așa cum spuneam, atributul action comunică browserului unde să trimită datele introduse în formular. Folosind comanda mailto: browserului i se indică să trimită datele la adresa de mail specificată în comandă.

Din păcate, această comandă nu este executată în acest mod decât de browserele Netscape. Browserele Internet Explorer nu recunosc comanda mailto: prezentă în cadrul formularelor. Acestea o execută la fel cum este ea executată atunci când face parte din eticheta <A> ca valoare a atributului href adică lansează aplicația de mail a utilizatorului.

Din acest motiv, inclusiv pentru trimiterea datelor la o adresă specificată de mail, se folosesc scripturi CGI. Pentru a nu intra încă în amănunte privind scripturile, vom folosi totuși în exemplele următoare comanda mailto:

Rețineți totuși că afișarea corectă a unui formular în pagină nu este decât prima etapă a utilizării formularului. Pentru a face o testare completă trebuie să aveți un script instalat pe server care să prelucreze datele din formular și să le transmită la adresa de mail specificată.

Pentru a defini elementele care fac parte din formular se utilizează etichetele <INPUT>, <SELECT>, <OPTION> și <TEXTAREA>.

3. Eticheta <INPUT>

Prin intermediul etichetei <INPUT> se pot introduce în formular câmpuri de editare (casete de text), câmpuri de tip *password*, butoane de expediere și anulare, butoane radio și casete de validare.

Eticheta <INPUT> are următoarele atribute:

type - prin care se precizează tipul elementului.

name - permite atașarea unui nume fiecărui element al formularului.

value - permite atribuirea unei valori inițiale unui element al formularului.

Perechea de atribute **name/value** (nume/valoare) este deosebit de importantă pentru buna funcționare a formularului.

Fiecare element introdus prin eticheta <INPUT> reprezintă o variabilă. Informația introdusă de utilizator în câmpul elementului respectiv reprezintă valoarea pe care o primește această variabilă. Din acest motiv, toate elementele introduse prin eticheta <INPUT> trebuie să aibă atribuit un nume. Acest lucru este obligatoriu deoarece, în caz contrar, variabila reprezentată de acel element nu va avea un identificator.

În plus, este indicat să atribuiți nume distincte diferitelor elemente care fac parte din formular. Altminteri, scriptul CGI care prelucrează datele nu va ști să facă distincția între variabile având același nume și valori diferite.

Atributul type poate avea următoarele valori:

- text - se folosește pentru a introduce un câmp de editare pe o singură linie
- password – se folosește pentru a insera un câmp de editare în care caracterele introduse sunt înlocuite cu asteriscuri (*)
- radio – folosit pentru a insera un șir de butoane radio (elemente care se folosesc pentru a selecta dintr-o listă de opțiuni una și numai una dintre ele)
- checkbox – folosit pentru a introduce un șir de casete de validare (elemente care se folosesc când dintr-o listă de opțiuni se poate alege mai mult de o singură variantă)
- submit – se folosește pentru a introduce un buton de expediere

- reset – se folosește pentru a introduce un buton de anulare a informațiilor introduse în formular
- button - se folosește pentru a introduce în formular un buton obișnuit
- hidden - se folosește pentru a introduce în formular un câmp ascuns

Fiecare dintre valorile atributului type generează un anumit tip de element în cadrul formularului.

Exemplul 11.1 creează un formular foarte simplu care conține un câmp de editare și un buton de expediere.

Exemplul 11.1

```
<HTML>
<HEAD>
<TITLE>formulare1</TITLE>
</HEAD>
<BODY>
<H1 align="center"> Caseta de text si buton de expediere</H1><HR>
<FORM action="mailto:adresa@yyy.com" method="post">
Introduceti adresa dvs. de mail:
<INPUT type="text" name="adresa" value="adresa mail"><BR>
<INPUT type="submit" value="expediaza">
</FORM>
</BODY>
</HTML>
```

După cum observați, numele atribuit câmpului de editare este "adresa" iar atributul value a primit ca valoare șirul de caractere "adresa mail". Acest text va fi afișat în interiorul câmpului de editare, înainte ca utilizatorul să înceapă introducerea informațiilor. Dacă atributul value nu era prezent sau era inițializat cu stringul vid (""), caseta de text ar fi fost goală.

În cadrul formularului este prezent și un buton de expediere, inserat de asemenea prin intermediul etichetei <INPUT> având atributul type setat cu valoarea "submit". Așa cum se observă din figură, pe buton este afișat textul "expediaza", deoarece pentru acest element, atributului value i-a fost atribuită ca valoare textul respectiv.

Casete de text

În Exemplul 11.2 am creat un formular care conține trei casete de text (câmpuri de editare) în care utilizatorul este rugat să introducă numele, prenumele și adresa de mail, precum și o casetă de tip *password*, care cere introducerea unei parole. O casetă de tip *password* este asemănătoare cu un câmp de editare, singura diferență fiind aceea că aici caracterele nu sunt afișate în clar ci sunt înlocuite cu asteriscuri.

În plus, formularul conține un buton de expediere și unul de anulare a datelor introduse.

Exemplul 11.2

```
<HTML>
<HEAD>
<TITLE>formulare2</TITLE>
</HEAD>
```

```

<BODY>
<H1 align="center">Casete de text</H1><HR>
<FORM action="mailto:adresa@yyy.com" method="post">
Introduceti numele:
<INPUT type="text" name="nume" value=""><BR>
Introduceti prenumele:
<INPUT type="text" name="prenume" value=""><BR>
Introduceti adresa de mail:
<INPUT type="text" name="mail" value=""><BR>
Introduceti parola:
<INPUT type="password" name="parola"><BR>
<INPUT type="reset" value="sterge">
<INPUT type="submit" value="expediaza">
</FORM>
</BODY>
</HTML>

```

Pentru elementele de tip casetă de text și *password* sunt utile și atributele:

maxlength – care stabilește numărul maxim de caractere care poate fi introdus în câmpul de editare

size – care stabilește lățimea acestui câmp

Butoane radio

În Exemplul 11.3 am creat un formular care conține un șir de butoane radio, prin care se cere părerea vizitatorului despre o pagină Web. După cum observați, fiecare element de tip buton radio are același nume și anume "opinie" deoarece nu se poate selecta decât un singur element, astfel încât orice ambiguitate este exclusă.

Fiecare buton radio trebuie să aibă o valoare. Din acest motiv pentru fiecare buton atributul *value* a fost setat la o altă valoare. Alegerea uneia dintre opțiuni, de exemplu "buna", face ca variabila "opinie" să primească valoarea "buna". Prin urmare, serverului îi va fi transmisă perechea "opinie=buna".

În plus, observați prezența atributului **checked**, care are rolul de a prestabili o anumită opțiune, pe care însă utilizatorul o poate schimba, dacă dorește.

Exemplul 11.3

```

<HTML>
<HEAD>
<TITLE>formulare3</TITLE>
</HEAD>
<BODY>
<H1 align="center">Butoane radio</H1><HR>
<FORM action="mailto:adresa@yyy.com" method="post">
Ce parere aveti despre aceasta pagina?<BR>
<INPUT type="radio" name="opinie" value="foarte buna" checked>Foarte buna<BR>
<INPUT type="radio" name="opinie" value="buna">Buna<BR>
<INPUT type="radio" name="opinie" value="destul de buna">Destul de buna<BR>
<INPUT type="radio" name="opinie" value="proasta">Proasta<BR>
<INPUT type="reset" value="anuleaza">
<INPUT type="submit" value="expediaza">

```

```
</FORM>
</BODY>
</HTML>
```

Casete de validare

Exemplul 11.4 creează un șir de casete de validare, care se deosebesc de butoanele radio prin faptul că se pot selecta mai multe opțiuni dintre cele prezentate

Exemplul 11.4

```
<HTML>
<HEAD>
<TITLE>formulare4</TITLE>
</HEAD>
<BODY>
<H1 align="center">Casete de validare</H1><HR>
<FORM action="mailto:adresa@yyy.com" method="post">
Care sunt hobby-urile dumneavoastra?<BR>
<INPUT type="checkbox" name="hobby" value="sport">Sport<BR>
<INPUT type="checkbox" name="hobby" value="film">Film<BR>
<INPUT type="checkbox" name="hobby" value="lectura">Lectura<BR>
<INPUT type="checkbox" name="hobby" value="jocuri">Jocuri pe computer<BR>
<INPUT type="checkbox" name="hobby" value="internet">Surfing pe net<BR>
<INPUT type="checkbox" name="hobby" value="alpinism">Alpinism<BR>
<INPUT type="submit" name="submit" value="expediaza">
</FORM>
</BODY>
</HTML>
```

O casetă de validare are două stări: marcată sau nemarcată (*on* sau *off*). Ca și la butoanele radio, atributul `name` are în exemplul considerat o singură valoare, "hobby", iar atributul `value` are, pentru fiecare casetă, valori distincte. Serverului îi sunt transmise numai acele valori care au fost marcate în casetele de validare corespunzătoare.

Și în cazul casetelor de validare este posibilă prezența atributului **checked** care preselecționează o anumită opțiune sau mai multe.

Butoane

În cadrul unui formular se pot introduce și butoane obișnuite (altele decât butoanele *Submit* și *Reset*). Rolul acestora este de a iniția anumite acțiuni în momentul când utilizatorul face click cu mouse-ul pe ele. Limbajul HTML nu oferă instrumentele necesare pentru a specifica ce anume se întâmplă când butonul este apăsat. Pentru aceasta este necesar să includeți în documentul HTML scripturi JavaScript care să trateze aceste evenimente.

Ca și butoanele *Submit* și *Reset*, butoanele obișnuite sunt incluse în formular cu ajutorul etichetei `<INPUT>` având atributul `type` setat la valoarea `button`:

```
<INPUT type="button" name="buton" value="text">.
```

Textul specificat ca valoare a atributului `value` va fi afișat pe buton.

Butoanele pot fi folosite pentru a valida informațiile introduse în formular, pentru a deschide documente și a iniția diverse alte acțiuni din partea browserului.

Câmpuri ascunse (hidden fields)

Câmpurile ascunse sunt elemente care fac parte dintr-un formular dar nu sunt vizibile în cadrul paginii. Prin intermediul acestora se pot include în formular informații care nu pot fi alterate de browser sau de utilizator. Introducerea unui astfel de câmp în formular se face prin eticheta

```
<INPUT type="hidden">.
```

Ca și în cazul celorlalte elemente ale formularului este necesară prezența atributelor name și value, ca în exemplul următor:

```
<INPUT type="hidden" name="ascuns" value="modifica">
```

Câmpurile ascunse pot servi la o gestionare mai bună a datelor transmise la server. De exemplu, să presupunem că avem un formular care cere utilizatorului câteva informații inițiale: numele și adresa. Pe baza acestora, aplicația de pe server inițiază afișarea unui nou formular care solicită introducerea unor informații mai detaliate.

Deoarece este plictisitor pentru vizitator să reia introducerea informațiilor inițiale, scriptul poate fi configurat să depună primele informații în câmpurile ascunse ale celui de-al doilea formular. În această situație este utilă folosirea câmpurilor ascunse deoarece serverul procesează un singur formular la un moment dat și nu are cum să știe că cel de-al doilea formular a fost completat de aceeași persoană.

4. Etichetele <SELECT> și <OPTION>

Cu ajutorul etichetei <SELECT> </SELECT> se poate introduce într-un formular un meniu derulant. Crearea unui meniu pentru vizitatorii paginii îi ajută la selectarea unor opțiuni dintr-o listă predefinită.

Fiecare opțiune care face parte din blocul <SELECT> se introduce prin eticheta <OPTION>.

În Exemplul 11.5 este construit un meniu derulant, din care vizitatorul poate selecta opțiunea dorită. Atributul **name** are același rol ca și în eticheta <INPUT>. Dacă vizitatorul selectează din meniu zodia Taur, de exemplu, la apăsarea butonului de expediere, serverului îi este transmisă perechea: "zodia=taur". Atributul **size** este setat la valoarea "1". Setarea atributului size la valoarea "1" creează o un meniu derulant cu o singură opțiune vizibilă inițial.

Exemplul 11.5

```
<HTML>
<HEAD>
<TITLE>formulare5</TITLE>
</HEAD>
<BODY>
<H1 align="center">Meniu derulant</H1><HR>
<FORM action="mailto:adresa@yyy.com" method="post">
Care este zodia dumneavoastra?<BR>
<SELECT name="zodia" size="1">
```

```

<OPTION>Berbec
<OPTION>Taur
<OPTION>Gemeni
<OPTION>Rac
<OPTION>Leu
<OPTION>Fecioara
<OPTION>Balanta
<OPTION>Scorpion
<OPTION>Sagetator
<OPTION>Capricorn
<OPTION>Varsator
<OPTION>Pesti
</SELECT>
<INPUT type="submit" value="expediaza">
</FORM>
</BODY>
</HTML>

```

Dacă atributul `size` este setat la o valoare mai mare decât 1 meniul va afișa un număr de opțiuni egal cu această valoare și va conține o bară de derulare pentru a face accesibile și celelalte opțiuni.

5. Eticheta <TEXTAREA>

Cu ajutorul etichetei <TEXTAREA> </TEXTAREA> puteți insera în pagină o casetă de text multi-linie care permite vizitatorului să introducă un text mai lung, care se poate întinde pe mai multe linii. Această etichetă se folosește atunci când dorim să cerem vizitatorului o opinie despre un anumit subiect, care necesită introducerea unui text mai lung decât o singură linie (acest element se mai numește și casetă de comentarii). <TEXTAREA> este o etichetă container, deci necesită eticheta de închidere </TEXTAREA>.

Atributele **rows** și **cols** stabilesc numărul de linii, respectiv de coloane rezervate pentru introducerea textului.

Exemplul 11.6 introduce în pagină o casetă pentru comentarii.

Exemplul 11.6

```

<HTML>
<HEAD>
<TITLE>formulare6</TITLE>
</HEAD>
<BODY>
<H1 align="center">Casetă de comentarii</H1><HR>
<FORM action="mailto:adresa@yyy.com" method="post">
Care este opinia dumneavoastra despre acest film?<BR>
<TEXTAREA name="comentariu" rows="10" cols="40"></TEXTAREA><BR>
<INPUT type="submit" value="expediaza">
</FORM>
</BODY>
</HTML>

```


Toate elementele prezentate pot fi reunite într-un singur formular sau puteți crea formulare diferite, în funcție de informațiile pe care doriți să le obțineți de la vizitatorii paginii.

Dacă doriți să introduceți în pagină mai multe formulare, sau elemente de tipuri diferite în cadrul aceluiași formular, trebuie să aveți în vedere aspectele care țin de alinierea acestora. O posibilitate de a alinia elementele conținute într-un formular este folosirea tabelelor.

Exemplul 11.7 conține elemente de mai multe tipuri încadrate într-un formular unic. Pentru alinierea unora dintre ele am folosit un tabel.

Exemplul 11.7

```
<HTML>
<HEAD>
<TITLE>formulare7</TITLE>
</HEAD>
<BODY>
<H1 align="center">Alinierea elementelor</H1><HR>
<FORM action="mailto:adresa@yyy.com" method="post">
<H3>Comandati Pizza</H3><P>
<P>Cate pizza doriti?
<INPUT name="nrpizza" value="0" size=3 maxlength=3><P>
<INPUT type="radio" name="marime" value="mare" checked>Mare<BR>
<INPUT type="radio" name="marime" value="medie">Medie<BR>
<INPUT type="radio" name="marime" value="normala">Normala
<P>Topping-uri<P>
<INPUT type="checkbox" name="topping" value="salam">Salam<BR>
<INPUT type="checkbox" name="topping" value="carnati">Carnati<BR>
<INPUT type="checkbox" name="topping" value="sunca">Sunca<BR>
<INPUT type="checkbox" name="topping" value="ciuperce">Ciuperce<BR>
<INPUT type="checkbox" name="topping" value="ceapa">Ceapa<BR>
<INPUT type="checkbox" name="topping" value="masline">Masline
<P>
<TABLE width="60%">
<TR>
<TD width="20%">Numele
<TD><INPUT type="text" name="nume">
<TR>
<TD width="20%">Telefonul
<TD><INPUT type="text" name="telefon">
<TR>
<TD width="20%">Adresa
<TD><TEXTAREA name="adresa" rows=6 cols=50></TEXTAREA>
<TR>
<TD width="20%">Numărul cartii de credit
<TD><INPUT type="password" name="creditcard" size=20>
</TABLE>
<P>
<CENTER>
<INPUT type="submit" value="Trimite comanda">
```

```
</CENTER>  
</FORM>  
</BODY>  
</HTML>
```

6. Validarea datelor

Unul dintre aspectele importante pe care trebuie să le aveți în vedere atunci când folosiți formulare în paginile dumneavoastră este validarea datelor introduse de vizitatorii lor. Limbajul HTML oferă relativ puține instrumente pentru îndeplinirea acestei sarcini. Totuși, aveți la dispoziție câteva posibilități.

Folosiți atributul `maxlength` atunci când introduceți un câmp de editare, pentru a împiedica vizitatorul să introducă un număr eronat de caractere. De exemplu, dacă cereți introducerea într-o casetă de text a codului numeric personal, acea casetă nu trebuie să permită introducerea a mai mult de 13 caractere, un câmp de editare care cere introducerea codului poștal nu trebuie să permită introducerea a mai mult de 4 caractere, etc.

Pentru a vă asigura că vizitatorul introduce date corecte într-un câmp de editare (de exemplu o adresă de mail validă trebuie să conțină obligatoriu caracterul "@") nu există instrumente HTML. Pentru a realiza astfel de sarcini sunt necesare scripturi JavaScript. În capitolul **JavaScript** veți găsi scripuri utile care fac posibile validările datelor, încă din momentul introducerii lor.

Folosiți de câte ori este posibil butoanele radio, casetele de validare și meniurile, pentru a simplifica procesul de introducere a datelor. Fiți cât mai explicit, asigurați indicațiile necesare pentru completarea formularelor, precizând, unde este cazul, dacă pot fi selectate mai multe opțiuni sau numai una singură.

Atenție!

Asigurați-vă că scriptul care prelucrează informațiile din formular are facilități de gestionare a erorilor.

Dacă este semnalată o eroare, furnizați vizitatorului un mesaj de eroare explicit și, eventual, câteva corecții posibile.

Una dintre cele mai frustrante situații este ca după completarea unui formular complex, la apariția unei erori, vizitatorul să fie nevoit să reia completarea integrală a formularului. Evitați acest lucru utilizând scripturi care returnează utilizatorului formularul completat, având câmpurile greșite marcate pentru corectare.

Nu uitați un amănunt foarte important: informațiile introduse de vizitatori în formulare vă sunt necesare dumneavoastră, pentru a îmbunătăți eficiența și calitatea site-ului sau chiar pentru a derula afaceri prin intermediul său. De aceea corectitudinea acestor date este esențială.

Faceți în așa fel încât vizitatorii site-ului dumneavoastră să completeze ușor și cu plăcere formularele prezente în pagini.

7. Rezumat

Formularele servesc la stabilirea unei legături interactive cu vizitatorii unei pagini Web și la obținerea de informații de la aceștia.

Un formular se introduce în pagină cu ajutorul etichetei `<FORM>` `</FORM>`.

În cadrul acestei etichete pot fi incluse diverse elemente care fac parte din formular:

- prin eticheta `<INPUT>` pot fi introduse câmpuri de editare, câmpuri de tip *password*, butoane radio, casete de selecție, butoane de expediere, butoane de anulare și butoane obișnuite
- prin eticheta `<SELECT>` pot fi introduse meniuri derulante, fiecare element al meniului fiind introdus prin eticheta `<OPTION>`
- prin eticheta `<TEXTAREA>` pot fi introduse câmpuri de editare multilinie.

Pentru a specifica unde vor fi trimise datele introduse în formular se folosește atributul `action` al etichetei `<FORM>`. Pentru a stabili modul în care aceste date sunt transmise se folosește atributul `method` al etichetei `<FORM>`.

Formularele reprezintă partea vizibilă a procesului de interacțiune cu vizitatorul paginii Web. Odată ce datele au fost introduse în câmpurile formularului, ele trebuie transmise, prelucrate și stocate, în funcție de nevoile creatorului paginii. Acest lucru se realizează cu ajutorul scripturilor CGI instalate pe server, despre care vom vorbi în capitolul următor.

8. Test

1. Ce atribut indică browserului unde să trimită informațiile din formular?

- a) `input`
- b) `action`
- c) `method`

2. Care dintre aceste valori nu aparține atributului `method`?

- a) `Get`
- b) `Href`
- c) `Post`

3. Care dintre următoarele afirmații este falsă?

- a) Prin eticheta `<INPUT>` se pot introduce în formular câmpuri de editare de tip *password*.
- b) Prin eticheta `<INPUT>` se pot introduce în formular butoane radio.
- c) Prin eticheta `<INPUT>` se pot introduce în formular meniuri derulante.

4. Pentru a introduce în formular un buton de expediere se folosește construcția:

- a) `<INPUT type="reset">`
- b) `<INPUT type="submit">`
- c) `<INPUT type="forward">`

5. Fie următoarea construcție:

```
<FORM action="mailto:adresa@yyy.com" method="post">
<INPUT type="radio" name="limba" value="romana">Limba romana
<INPUT type="radio" name="limba" value="germana">Limba germana
<INPUT type="radio" name="limba" value="maghiara">Limba maghiara
</FORM>
```

Dacă este bifată opțiunea Limba romana, care este perechea nume/valoare care va fi transmisă serverului?

- a) "limba=Limba romana"
- b) "limba=romana"
- c) "radio=romana"

6. Care este diferența dintre un câmp de editare de tip text și unul de tip password?

- a) Într-un câmp de tip text se pot introduce mai multe linii, într-unul de tip *password* o singură linie.
- b) Într-un câmp de tip text caracterele introduse sunt afișate, într-unul de tip *password* sunt înlocuite cu asteriscuri.
- c) Câmpul de editare se introduce cu ajutorul etichetei <INPUT>, câmpul de tip *password* se introduce cu ajutorul etichetei <SELECT>.

7. Pentru a insera într-un formular o casetă de validare se folosește eticheta <INPUT> cu atributul type setat la valoarea:

- a) "radio"
- b) "checkbox"
- c) "option"

8. Ce realizează eticheta <SELECT>?

- a) Introduce un buton de anulare.
- b) Introduce un meniu derulant.
- c) Introduce o listă de casete de validare.

9. La ce servește eticheta <TEXTAREA>?

- a) Stabilește zona din fereastra browserului care este rezervată pentru text.
- b) Introduce în formular un câmp de editare multi-linie.
- c) Introduce în formular un câmp de editare pe o singură linie.

10. Care dintre următoarele afirmații este falsă?

- a) Datele introduse în formular sunt trimise la server, dacă este folosit un script CGI.
- b) Datele introduse în formular sunt trimise la o adresă de mail specificată, dacă este folosită comanda mailto:
- c) Datele introduse în formular sunt stocate pe hard-disk-ul vizitatorului paginii.