Basi di dati - Laboratorio

Corso di Laurea in Bioinformatica

Docente: Barbara Oliboni

Lezione 5

Contenuto della lezione

- Il Linguaggio HTML (HyperText Markup Language)
  - Struttura del documento
  - Formattazione del testo
  - Collegamenti ipertestuali
  - Immagini
  - Tabelle
  - Form
- Il protocollo HTTP (Hypertext Transfer Protocol)

HyperText Markup Language

- Linguaggio di descrizione di testi secondo lo schema SGML (Standard Generalized Markup Language)
- Gli ipertesti del Web sono scritti in HTML
- HTML non è un linguaggio di programmazione
- HTML non è "case sensitive": non distingue i caratteri minuscoli da quelli maiuscoli all'interno dei TAG.
- HTML è un linguaggio di marcatura che permette di descrivere come il contenuto di un documento verrà presentato

File HTML

- Un documento HTML è un file in formato testo che ha estensione .html o .htm
- Il file HTML che contiene un documento è formato dal contenuto del documento più la marcatura
- La marcatura descrive il modo in cui il contenuto verrà presentato
- File HTML = contenuto + marcatura

File HTML (2)

- I documenti HTML si possono creare con degli editor di testo
  - Se si usa Word ad esempio è possibile salvare il documento con estensione .html e trasformare la formattazione del testo in tag HTML con l'opzione "solo testo con interruzione di riga"
- I browser leggono i documenti HTML e li visualizzano interpretando le specifiche di formattazione (marcatura)

HTML: concetti generali

- La marcatura prevede l'uso di etichette dette TAGS
- I TAG racchiudono il testo di cui definiscono la formattazione

  <tag> testo </tag>
- Il significato di un tag può essere modificato tramite attributi

  <tag attributo=valore> testo </tag>

Struttura del documento

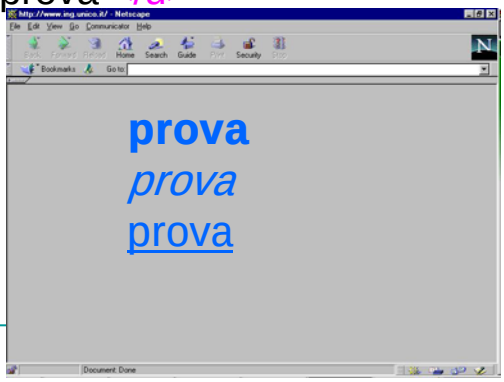
- File HTML, struttura generale:
  <html> intestazione + corpo </html>
- Intestazione: <head> … </head>
  contiene informazioni sul documento:
  titolo <title>… </title>
- Corpo: <body> … </body>
  contiene il testo del documento e i tag per la presentazione

Struttura del documento: TAG

```
<HTML>
  <HEAD>
    <TITLE>
    </TITLE>
  </HEAD>
  <BODY>
  </BODY>
</HTML>
```

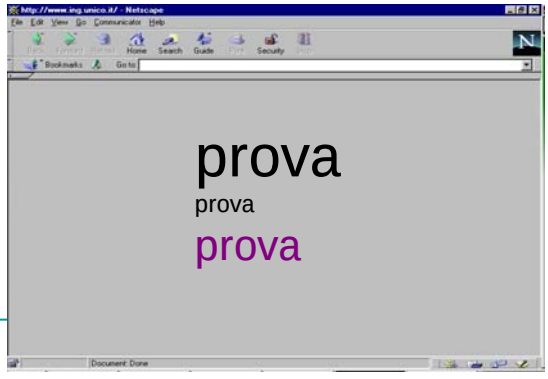
Formattazione del testo

- grassetto prova
- corsivo prova
- sottolineato <u> prova </u>



Formattazione del testo

- Dimensioni: prova
   prova
- Colore: prova



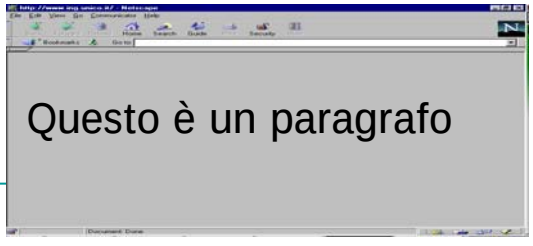
Titoli

- I livelli di titolazione sono 6:
  - Livello 1 (massimo) <h1> Titolo livello 1 </h1>
  - Livello 2 <h2> Titolo livello 2 </h2>
  - …
  - Livello 6 (minimo) <h6> Titolo livello 6 </h6>

Paragrafi

In HTML il comando “Invio” non ha significato: il browser legge la sequenza di parole senza badare alle interruzioni di linea.

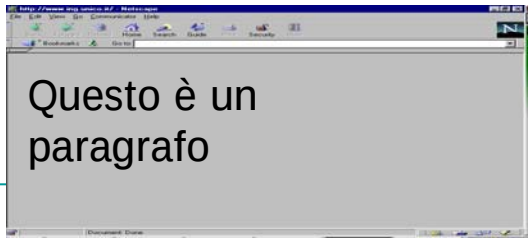
Paragrafi: Questo è un paragrafo



Interruzione di linea

Per interrompere una linea in un punto desiderato si usa il TAG
:

```
<p>Questo è un <br>paragrafo</p>
```



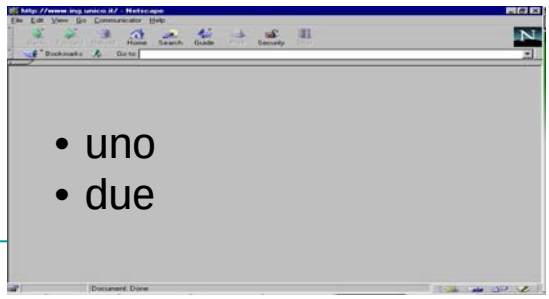
Testo formattato

Per rendere visibili spazi aggiunti nel documento HTML ed interruzioni di linea si usa:

```
<pre>prova
        prova</pre>
```

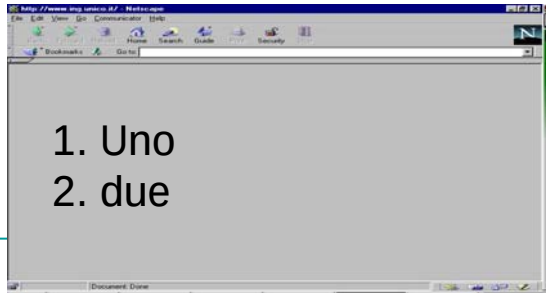
Liste non numerate

```
<ul>
  <li> uno </li>
  <li> due </li>
</ul>
```



Liste numerate

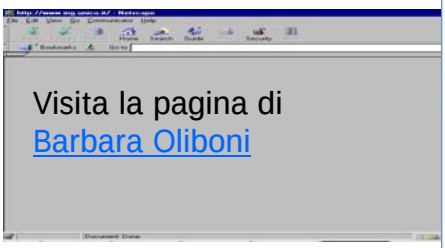
```
<ol>
  <li> uno </li>
  <li> due </li>
</ol>
```

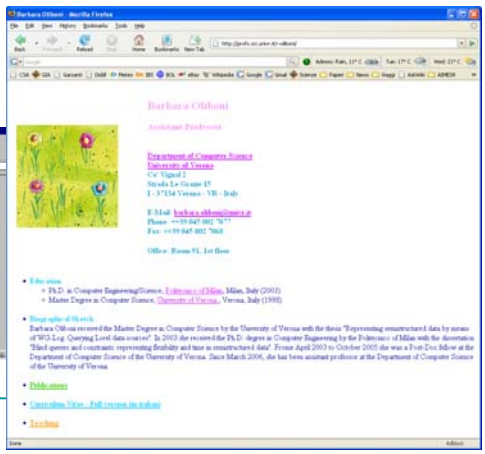


Collegamenti ipertestuali verso altri documenti

Visita la pagina di

Barbara Oliboni

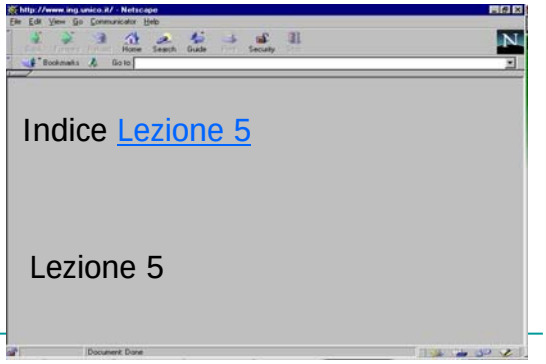




Collegamenti ipertestuali sullo stesso documento

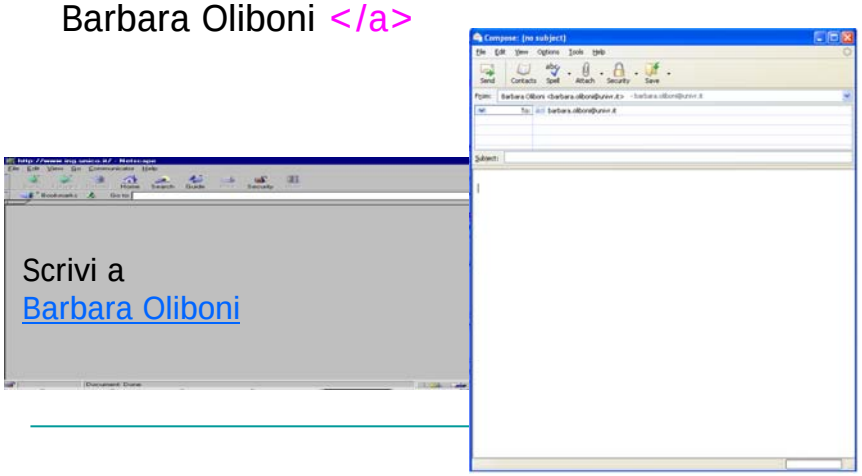
Indice Lezione 5

 Lezione 5



Collegamenti ipertestuali

Scrivi a
Barbara Oliboni

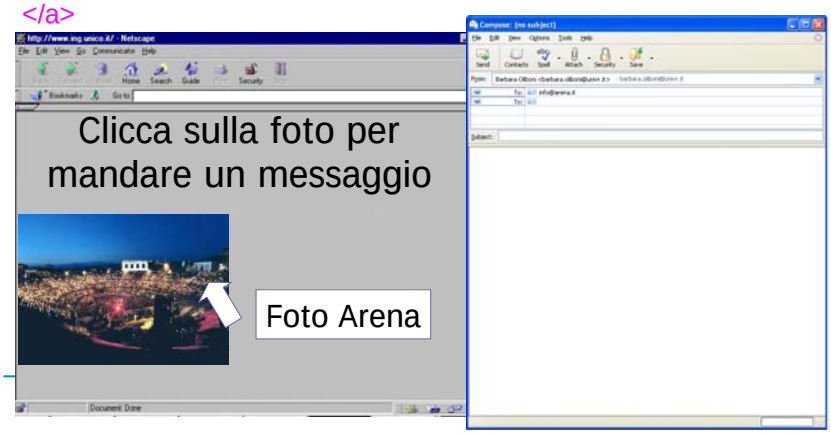


Immagini

 Foto dell’Arena di Verona
<img src = “arena.jpg” width=“200” height=“400”
alt=“Foto Arena”>

Immagini + collegamenti

```
<p> Clicca sulla foto per mandare un messaggio </p>
<a href = mailto:info@arena.it >
  <img src = “arena.jpg” width=“200” height=“400” alt=“Foto Arena”>
</a>
```



Tabelle

- Per definire una tabella:
  <TABLE> … </TABLE>
- Per definire la didascalia della tabella (o titolo):
  <CAPTION> … </CAPTION>
- Per specificare una riga dentro la tabella:
  <TR> … </TR>
- Per definire una cella di intestazione:
  <TH> … </TH>
- Per definire una cella per i dati:
  <TD> … </TD>

Tabelle: esempio 1

```
<TABLE border="1" >
 <CAPTION> Risultati esame </CAPTION>
 <TR>
    <TH>Nome</TH>
    <TH>Voto</TH>
 </TR>
 <TR>
    <TD>Mario Rossi</TD>
    <TD>28</TD>
 </TR>
 <TR>
    <TD>Lucia Verdi</TD>
    <TD>30</TD>
 </TR>
</TABLE>
```

Tabelle: risultato esempio 1

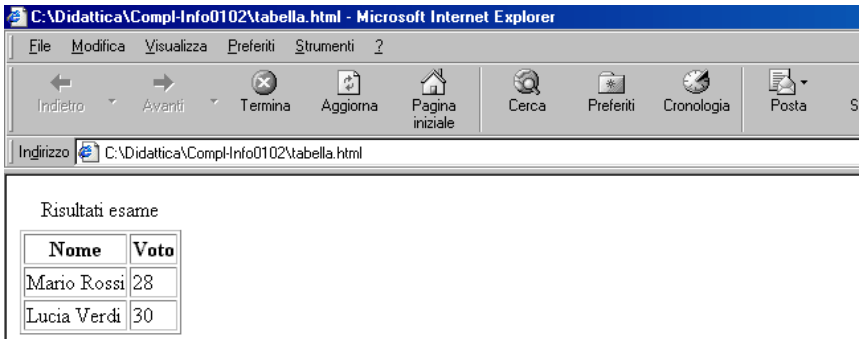
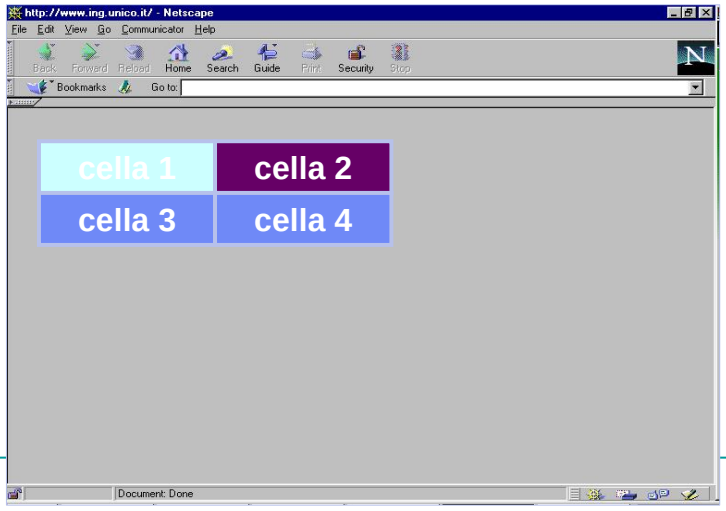


Tabelle: esempio 2

```
<TABLE border="1" width="50%" bgcolor="#FFFF00">
 <TR>
   <TD width="50%" bgcolor="#0000FF"> cella 1</TD>
   <TD width="50%"> cella 2</TD>
 </TR>
 <TR bgcolor="#C0C0C0">
    <TD width="50%">cella 3</TD>
   <TD width="50%">cella 4</TD>
 </TR>
</TABLE>
```

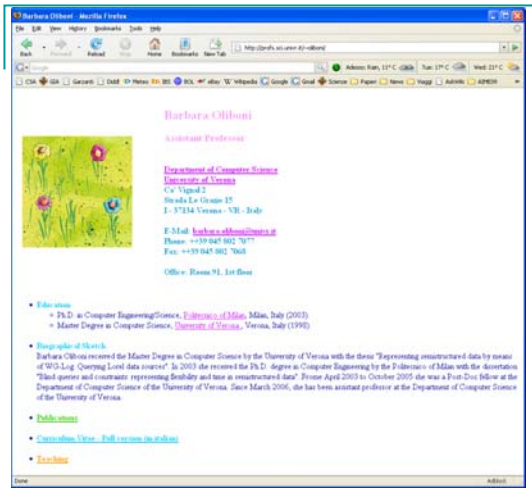
Tabelle: risultato esempio 2



Lettere accentate

- à à
- è è
- ì ì
- ò ò
- ù ù
- é è

- Esempio:
  Il giudizio è
  più che buono
  → Il giudizio è più che buono



```
<HTML>
    <HEAD>
        <TITLE> Barbara Oliboni </TITLE>
    </HEAD>
    <body text="#000099" bgcolor="#FFFFFF" link="#CC00FF" vlink="#CC00FF">
```

```
<table CELLPADDING=10 COL="0" >
    <tr>
        <td>
         <img SRC="./imgs/fiori.jpg" ALIGN=CENTER>
        </td>

        <td>
        <blockquote>

        <font color="#FF99FF"><h2>Barbara Oliboni</h2></font>
        <font color="#FF99FF"><h3>Assistant Professor</h3></font>
        <br>
        <font color="#0099CC"><b>
        <a href="http://www.di.univr.it/">Department of Computer Science</a><br>
        <a href="http://www.univr.it">University of Verona</a><br>
                Ca' Vignal 2<br>Strada Le Grazie 15<br>I - 37134 Verona - VR - Italy
        <br><br>
        E-Mail: <a href="mailto:barbara.oliboni@univr.it">barbara.oliboni@univr.it</a><br>
        Phone: ++39 045 802 7077<br>
        Fax: ++39 045 802 7068
        <br><br>
        Office: Room 91, 1st floor
        </b></font>
        </blockquote>
        </td>
    </tr>
</table>
```

```
<ul>
    <li><font color="#00CCFF"><b>Education</b></font>
    <ul>
        <li>Ph.D. in Computer Engineering/Science, <a href="http://www.polimi.it">Politecnico of
                Milan</a>, Milan, Italy (2003).</li>
        <li>Master Degree in Computer Science, <a href="http://www.univr.it">
        University of Verona </a>, Verona, Italy (1998).</li>
    </ul>

    <br>
    <li><font color="#00CCFF"><b>Biographical Sketch</b></font><br>

    Barbara Oliboni received the Master Degree in Computer Science by
    the University of Verona with the thesis "Representing semistructured
    data by means of WG-Log: Querying Lorel data sources".
    In 2003 she received the Ph.D. degree in Computer Engineering by
    the Politecnico of Milan with the dissertation "Blind queries and
    constraints: representing flexibility and time in semistructured data".
    Frome April 2003 to October 2005 she was a Post-Doc fellow at the
    Department of Computer Science of the University of Verona.
    Since March 2006, she has been assistant professor at the
    Department of Computer Science of the University of Verona.
```

```
<br><br>
<li><A HREF="papers.html">
      <FONT COLOR="33CC00">
      <b>Publications</b>
      </FONT></A>

<br><br>
<li><A HREF="./CV/cv_Oliboni.pdf">
      <FONT COLOR="00CCFF">
      <b>Curriculum Vitae - Full version (in italian)</b><
      /FONT></A>

<br><br>
<!-- questo  un commento -->
<li> <A HREF="teaching.html">
      <FONT COLOR="FF9900"><b>Teaching</b></FONT></A>
</ul>

</font>

</BODY>
</HTML>
```

FORM HTML

- Una FORM consente di specificare, all'interno di un documento HTML, una sezione in cui l'utente può inserire dei dati.
- La sintassi del tag <FORM> è la seguente:

```
<form
    action = "URI"
    method = "metodo"
    enctype = "tipo-contenuti"
    accept-charset = "set-di-caratteri"
    accept = "tipi-di contenuti"
    name = "nome-modulo" >
...
</form>
```

Attributi usati raramente

Attributo ACTION

- Unico attributo necessario.
- Nell'attributo action viene specificato l'URI del programma di elaborazione: quando i dati sono stati inseriti e l'utente seleziona il pulsante di invio, il browser Web crea una richiesta HTTP contenente tutti i dati e la invia al programma di elaborazione.
- Il programma di elaborazione può essere ad esempio una Servlet.

Attributo METHOD

- Utilizzato per indicare il tipo di richiesta:
  - GET: i valori della FORM vengono aggiunti all'identificatore URI della richiesta sotto forma di stringa.
    - Problemi:
      - I valori introdotti sono visibili come coppie nome/valore nella riga degli indirizzi del Browser e nei file di registrazione del server Web. Quindi GET risulta inadatto per inviare dati riservati.
      - Alcuni server e browser possono prevedere restrizioni sulla lunghezza dell'indirizzo che può essere inviato.
  - POST: i valori della FORM vengono forniti nello stream di input.
- L'attributo method è opzionale: se non viene specificato viene usato il metodo GET.

Elementi di INPUT

- Nel corpo di una FORM vengono descritti i vari campi di input.
- Per creare elementi di input vengono utilizzati quattro tipi di tag HTML:
  - <INPUT>: tag generico
  - <SELECT> e <OPTION>: per creare un menu a tendina (o casella a discesa)
  - <TEXTAREA>: per le caselle di testo multiriga
  - <BUTTON>: per creare pulsanti

<INPUT>

```
<INPUT
   name = “nome del campo”
   type = “[text | password | checkbox |
            radio | hidden | submit | reset]”
   value = “valore iniziale”
   size = “dimensione”
   maxlength = “numero massimo di
                  caratteri” >
```

Attributi di <INPUT>

- name: utilizzato per assegnare un identificatore al campo
- type: indica il tipo del campo. Se non viene specificato si presume sia TEXT
- value: può essere utilizzato per assegnare un valore iniziale al campo
- size: indica le dimensioni del campo in pixel o in caratteri (per i campi di testo)
- maxlength: indica il numero massimo di caratteri che possono essere digitati nel campo

Il controllo TEXT

- Per introdurre un'unica riga di input.
- Sintassi:

```
<INPUT
    type = "text"
    value = "valore-iniziale"
    size = "dimensione"
    maxlength = "numero massimo di caratteri" >
```

- Esempio.

  Testo HTML:

  <input name="testo" type="text" value="testo iniziale" size="15">

  Resa del browser: 

Il controllo PASSWORD

- Variante del controllo TEXT.
- I caratteri introdotti non vengono visualizzati, ma vengono mascherati da *
- Sintassi:

```
<INPUT
    type = "password"
    value = "valore-iniziale"
    size = "dimensione"
    maxlength = "numero massimo di caratteri" >
```

- Esempio.

  Testo HTML:

  <input name="passwd" type="password" value="1g%34D9$" size="15" maxlength="10">

  Resa del browser: 

Il controllo CHECKBOX

- Casella di controllo che consente di presentare un'opzione che può essere vera o falsa.
- Sintassi:

```
<INPUT
    type = "checkbox"
    name = "nome"
    value = "valore-iniziale"
    checked >
```

- Esempio.

  Testo HTML:

```
A<input name="scelta1" type="checkbox" value="A">
B<input name="scelta2" type="checkbox" value="B" checked>
```

  Resa del browser: A ☐ B ☑

Il controllo RADIO

- Pulsante di selezione che consente di presentare un'opzione che può essere vera o falsa.
- Il funzionamento del pulsante è mutuamente esclusivo.
- Sintassi:

```
<INPUT
    type = "radio"
    name = "nome"
    value = "valore-iniziale"
    checked >
```

- Esempio.

  Testo HTML:

```
A<input name="opzione_esclusiva" type="radio" value="A" checked>
B<input name="opzione_esclusiva" type="radio" value="B">
```

  Resa del browser: A ◉ B ○

Il controllo HIDDEN

- Campo non visualizzato.
- Utilizzato per creare un parametro di valore costante (esempio codice che una Servlet potrà utilizzare come chiave per accedere ad una tabella)
- Sintassi:
  <INPUT
  type = “hidden”
  value = “valore-iniziale” >

- Esempio.
  Testo HTML:
  <input name="variabile_nascosta" type="hidden" value="1234">

  Resa del browser:

Il controllo SUBMIT

- Per inviare i dati inseriti al server.
- Sintassi:
  <INPUT
  type = “submit”
  value = “valore-iniziale” >

- Esempio.
  Testo HTML:
  <input type="submit“ value=“Invia query”>

  Resa del browser: 

Il controllo RESET

- Per riportare tutti i controlli al valore iniziale.
- Sintassi:

```
<INPUT
    type = “reset”
    value = “valore-iniziale” >
```

- Esempio.

  Testo HTML:

  <input type="reset" value="Annulla">

  Resa del browser: 

Attributo TYPE

Attributo Type	Resa del Browser	Testo HTML
text		**<input name="testo" type="text" value="testo iniziale" size="15">**
password		**<input name="passwd" type="password" value="1g%34D9$" size="15" maxlength="10">**
checkbox		**A<input name="scelta1" type="checkbox" value="A"> B<input name="scelta2" type="checkbox" value="B" checked>**
radio		**A<input name="opzione_esclusiva" type="radio" value="A" checked> B<input name="opzione_esclusiva" type="radio“ value="B">**
hidden		**<input name="variabile_nascosta" type="hidden" value="1234">**
submit		**<input type="submit“ value=“Invia query”>**
reset		**<input type="reset" value="Annulla">**

<SELECT> e <OPTION>

```
<SELECT
   name = “nome”
   size = “numero di elementi visibili” [multiple]>
   <OPTION
        value = “valore”
        [selected]>
   </OPTION>
   ...
</SELECT>
```

Attributi di <SELECT> e <OPTION>

- SELECT:
  - name: utilizzato per assegnare un nome al controllo
  - size: indica il numero di elementi visibili contemporaneamente, ovvero l’altezza del menu.
  - multiple: consente all’utente di selezionare più elementi
- OPTION:
  - value: specifica il valore restituito quando viene selezionato un certo elemento.
  - selected: se presente, preseleziona l’elemento.

<SELECT>: esempio 1

Codice HTML:

```
<select name="lista">
    <option value="blue">blue</option>
    <option value="rosso" selected>rosso</option>
    <option value="verde">verde</option>
    <option value="giallo">giallo</option>
    <option value="bianco">bianco</option>
    <option value="nero">nero</option>
</select>
```

Resa del browser:

<SELECT>: esempio 2

Codice HTML:

```
<select name="lista" multiple>
    <option value="blue">blue</option>
    <option value="rosso" selected>rosso</option>
    <option value="verde">verde</option>
    <option value="giallo" selected>giallo</option>
    <option value="bianco">bianco</option>
    <option value="nero">nero</option>
</select>
```

Resa del browser:

<TEXTAREA>

```
<TEXTAREA
   name = “nome”
   rows = “numero di righe”
   cols = “numero di colonne”>
   testo
   </TEXTAREA>
```

Attributi di <TEXTAREA>

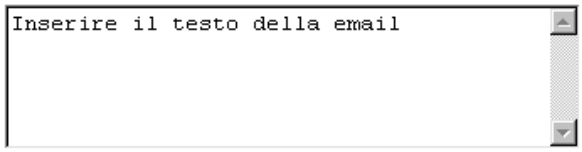
- name: assegna il nome al campo
- rows: specifica il numero di righe visualizzate nella finestra del browser per l’area di testo
- cols: specifica la larghezza in caratteri dell’area di testo da visualizzare.

<TEXTAREA>: esempio 1

Codice HTML:

```
<textarea name="email" rows="5" cols="40">
        Inserire il testo della email
</textarea>
```

Resa del browser:



Il Protocollo HTTP

- HTTP (Hypertext Transfer Protocol) è il "linguaggio" utilizzato per controllare l'invio di documenti HTML via Internet.
- Il protocollo HTTP prescrive le regole mediante le quali i browser effettuano le richieste e i server forniscono le relative risposte.
- Documentazione: RFC 2616 (http://www.freesoft.org/CIE/RFC/index.htm) versione aggiornata delle specifiche del protocollo HTTP versione 1.1.

La richiesta HTTP

- HTTP è un protocollo *senza stati* a richieste e risposte.
- Senza stati significa che il server Web non ricorda nulla delle richieste pervenute in precedenza dallo stesso client: il protocollo considera semplicemente la richiesta attuale di un documento e la risposta costituita dal documento stesso.

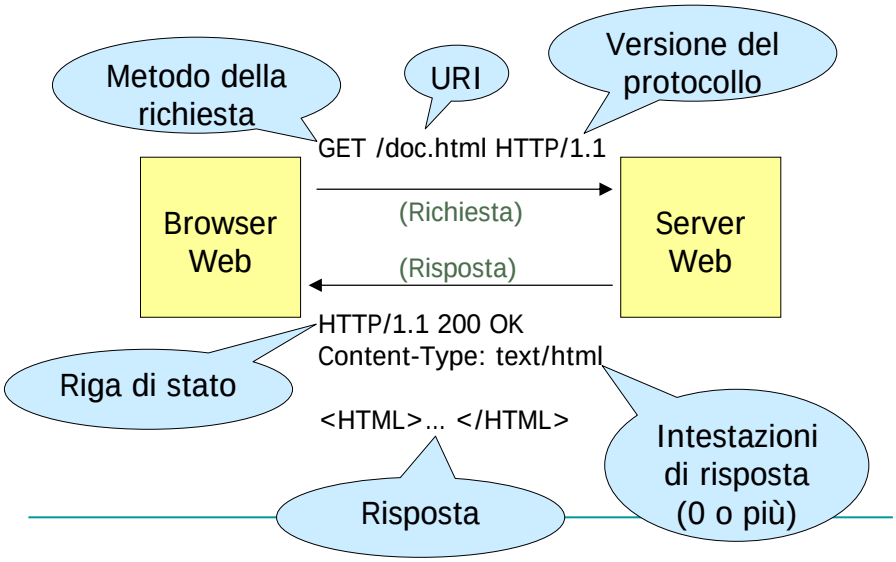
La richiesta HTTP (2)

- Operazioni di base:
  1. Un'applicazione client (browser Web) apre una connessione verso la porta HTTP del server Web (normalmente la porta 80).
  2. Il client invia una richiesta attraverso la connessione aperta.
  3. Il server Web analizza la richiesta ed individua la risorsa specificata.
  4. Il server invia una copia della risorsa.
  5. Il server chiude la connessione.

Connessione al Server Web

- Normalmente un server Web riceve le richieste sulla porta 80, in questo caso l'indirizzo http://profs.sci.univr.it/~oliboni/index.html fa riferimento al documento ~oliboni/index.html sul server Web in esecuzione sull'host profs.sci.univr.it e operante sulla porta standard 80.
- Se invece il server Web utilizzasse la porta 8080, l'indirizzo dovrebbe essere: http://profs.sci.univr.it:8080/~oliboni/index.html

Funzionamento di HTTP



Esempio

- Sulla riga di indirizzo del browser viene digitato http://profs.sci.univr.it/~oliboni/index.html
- Il browser web apre una connessione sulla porta 80 del server web profs.sci.univr.it
- Il browser web scrive la riga
  GET ~oliboni/index.html HTTP/1.0
  seguita da una riga vuota

Esempio (2)

- Il server web restituisce la risposta:

```
HTTP/1.1 200 ok
Date: Mon, 31 Mar 2003 14:27:43 GMT
…
Content-Length: 1619
Content-Type: text/html

<HTML>
<HEAD>
<TITLE> Barbara Oliboni </TITLE>
</HEAD>
<BODY>

        …
</BODY>
</HTML>
```

Esempio (3)

- Il browser analizza la riga di stato e trova il codice di stato 200 ok che indica che la richiesta ha avuto successo.
- Il browser analizza le intestazioni di risposta che indicano che verranno inviati 1619 byte di codice HTML.
- Il browser legge il codice HTML e visualizza il risultato.

- Se il codice HTML contiene riferimenti ad altre risorse che devono essere caricate con il documento, allora il browser invia una richiesta per ogni risorsa necessaria.