

Tecnologie Web: HTML5 Avanzato

Prof. Raffaele Montella, Ph.D.

raffaele.montella@uniparthenope.it

Moduli di input (Forms)

- L'HTML5 non serve solo per la fruizione passiva di contenuti
- L'utente può inviare al server web dei dati
- Si usano moduli di input detti form
 - I form sono costituiti da elementi detti controlli
 - I form inviano dati al server interagendo con risorse capaci di elaborarli e restituire un risultato

Moduli di input (Forms)

- Le risorse capaci di elaborare i dati inviati dai form sono programmi che funzionano sul server
- Attraverso i form è possibile definire l'interfaccia utente delle applicazioni web
- I controlli dei form possono avere un funzionamento complesso grazie allo scripting lato client

Moduli di input (Forms)

- `<form>`: block
 - Definisce il form costituito da controlli immersi nel codice HTML5
 - Attributi:
 - `method`: GET|POST, seleziona il verbo HTTP da usare per l'invio di dati tramite una URI
 - `action`: specifica la risorsa che elaborerà i dati una volta che la form è stata sottomessa (inviata) al server
 - `enctype`: specifica come codificare i dati
- Esempio:
`<form method="get" action="http://...">`
...
`</form>`

Moduli di input (Forms)

- `<input>`: inline
 - Assume diverse forme a seconda dell'attributo `type`:
 - `text`: controllo per l'inserimento del testo
 - `password`: come `text`, ma visualizza * invece che i caratteri
 - `checkbox`: un elemento che può essere selezionato o meno
 - `button`: un pulsante da premere
 - `radio`: un elemento che può essere selezionato in maniera mutuamente esclusiva rispetto ad un gruppo
 - `submit`: come `button`, ma invia la form al server invocando la risorsa specificata in `action`

Moduli di input (Forms)

- `<input>`: inline
 - Assume diverse forme a seconda dell'attributo type:
 - reset: cancella eventuali dati già immessi (e non inviati)
 - file: permette la selezione di un file sul client
 - hidden: non ha rendering grafico, ma permette di trasferire al server dati non visibili sulla pagina
 - image: pulsante con icona
 - Attributi:
 - id: nome dell'elemento
 - value: valore iniziale dell'elemento

Moduli di input (Forms)

- `<select>`, `<option>`: inline
 - Menu a tendina (drop down list)
 - Ogni voce è definita dall'elemento `<option>`
 - Ogni voce può essere abilitata e o visibile
- `<textarea>`: inline
 - Area di editing multilinea di dimensioni cols e rows specificata come attributo

Moduli di input (Forms)

- `<button>`: inline
 - Controllo a pulsante
 - Assume forme diverse a seconda dell'attributo `type`:
 - `submit`: invia la form al server
 - `reset`: cancella il contenuto dei campi
 - `button`: pulsante generico
- `<fieldset>`: block
 - Raggruppa controlli semanticamente correlati

Moduli di input (Forms)

- `<label>`
 - Definisce un'etichetta per il controllo il cui valore dell'attributo `id` è uguale a quello dell'attributo `for` di `<label>`
- `<label>`
 - Legenda del form
- `<optgroup>`
 - Elemento di raggruppamento delle opzioni

Moduli di input (Forms)

- Esempio 1:

```
<form action="http://..." method="post" id="form1">
  <p>
    <label for="altezza">Altezza</label>
    <input type="text" id="altezza" name="altezza" size="10" maxlength="3" />
  </p>
  <p>
    <label for="peso">Peso</label>
    <input type="text" id="peso" name="peso" size="10" maxlength="3"/>
  </p>
  <p>
    <input type="submit" value="Spedisci"/>
    <input type="reset" value="Annulla"/>
  </p>
</form>
```

Moduli di input (Forms)

- Esempio 2:

```
<form action="#" method="post" id="form1">
  <p>
    <label for="altezza">Altezza</label>
    <input type="text" id="altezza" size="10" maxlength="3" />
  </p>
  <p>
    <label for="peso">Peso</label>
    <input type="text" id="peso" size="10" maxlength="3"/>
  </p>
  <p>
    <label for="F">Femmina</label>
    <input type="radio" name="sesso" id="F" value="F"/>

    <label for="M">Maschio</label>
    <input type="radio" name="sesso" id="M" value="M"/>

    <label for="N">Non dichiarato</label>
    <input type="radio" name="sesso" id="N" value="N" checked="checked"/><br/>
  </p>
  <p>
    <input type="submit" value="Spedisci"/>
    <input type="reset" value="Annulla"/>
  </p>
</form>
```

Tabelle (Tables)

- Il modulo Tables descrive gli elementi necessari alla gestione delle tabelle
- Le tabelle sono usate per il posizionamento degli elementi nella pagina: è un uso improprio!
- Usare l'elemento `<div>` ed i fogli di stile CSS
- Le tabelle sono organizzate in righe
- Ciascuna riga è divisa in celle
- Il numero di celle può non essere costante per ogni riga

Tabelle (Tables)

- `<caption>`: inline
 - Legenda della tabella
- `<table>`: block
 - Comprende gli altri elementi della tabella
 - Può avere una `<caption>`
 - Può contenere un elemento `<tr>` per ciascuna riga della tabella
 - Può contenere gli elementi `<thead>`, `<tfoot>` e `<tbody>` per meglio specificare il contenuto della tabella dal punto di vista semantico
- `<tr>`: block
 - Riga della tabella
 - Contiene gli elementi `<th>` e `<td>`

Tabelle (Tables)

- `<td>`: block
 - Cella della tabella
 - Può contenere gran parte degli elementi HTML5 compreso `<table>`
- `<th>`: block
 - Intestazione delle colonne della tabella
 - Come `<td>`, ma con un rendering differente
- `<col>`, `<colgroup>`
 - Gestione dei gruppi di colonne
 - `<colgroup>` contiene 0 o più `<col>`
- `<thead>`, `<tfoot>`, `<tbody>`
 - Se usati contengono almeno una riga
 - La riga diventa parte integrante dell'intestazione, del piede e del corpo della tabella

Tabelle (Tables)

- Esempio 1: Semplice tabella con titolo

```
<table>  
  <caption>Tabella Semplice</caption>  
  <tr>  
    <th>Riga 1</th>  
    <td>A1</td><td>B1</td><td>C1</td>  
  </tr>  
  <tr>  
    <th>Riga 2</th>  
    <td>A2</td><td>B2</td><td>C2</td>  
  </tr>  
</table>
```

Tabelle (Tables)

- Esempio 2: Celle estese su più righe e colonne

```
<table>  
  <tr>  
    <th>Riga 1</th>  
    <td rowspan="2">A1</td>  
    <td colspan="2">B1</td>  
  <tr>  
  <tr>  
    <th>Riga 2</th>  
    <td>A2</td><td>B2</td>  
  </tr>  
</table>
```


Tabelle (Tables)

- Esempio 3: Tabelle complesse

```
<table>  
  <caption>Le famiglie di Springfield</caption>  
  <thead>  
    <tr>  
      <th>Famiglia</th>  
      <th>Figli</th>  
    </tr>  
  </thead>  
  <tfoot>  
    <tr>  
      <th>Famiglia</th>  
      <th>Figli</th>  
    </tr>  
  </tfoot>  
  ...
```

Tabelle (Tables)

- Esempio 3: Tabelle complesse

...

```
<tbody>
  <tr>
    <td>Simpson</td>
    <td>Bart, Lisa, Maggie</td>
  <tr>
    <td>Flanders</td>
    <td>Rod, Tod</td>
  </tr>
  <tr>
    <td>Van Houten</td>
    <td>Milhouse</td>
  </tr>
  <tr>
    <td>Gamble</td>
    <td></td>
  </tr>
</tbody>
</table>
```

Fogli di Stile

- Cascade Style Sheet
- I fogli di stile sono documenti che definiscono la modalità di rendering di ogni elemento presente nella pagina web
- Sono detti “cascade” perché le caratteristiche di visualizzazione possono essere definite in maniera gerarchica
- In HTML5 esistono due metodi per collegare i fogli di stile ad una pagina web
 - Regole di stile all'interno dell'elemento `<style>`
 - Collegamento ad un file esterno attraverso l'elemento `<link>` contenuto in head

Fogli di Stile

- Esempio 1: <style>
 <head>
 <title>Titolo della pagina</title>
 <style type="text/css">
 p {
 color: red;
 font-size: 14pt;
 }
 </style>
 ...
 </head>
- Il contenuto dell'elemento <style> segue la sintassi dei CSS

Fogli di Stile

- Esempio 2: <link>
 <head>
 <title>Titolo della pagina</title>
 <link rel="stylesheet" href="pagina.css" type="text/css"/>
 ...
 </head>
- Il file pagina.css contiene il foglio di stile che viene applicato alla pagina.
- Delle due forme quest'ultima è quella più conveniente

Altri Elementi

- Meta Informazioni
 - Elementi <meta> inclusi nell'elemento <head>
 - Definizione delle parole chiave
 - Redirect/refresh della pagina lato client
- Linguaggi di scripting
 - Elemento <script>
 - Contiene codice interpretato lato client
 - Contiene il riferimento ad una risorsa che offre tale codice

Altri Elementi

- Inclusione di altri componenti
 - Elementi `<object>` e `<param>`
 - Servono ad includere, ad esempio, un componente flash o un player multimediale
 - L'elemento `<param>` serve per selezionare i valori dei parametri necessari al componente incluso
- Eventi
 - Attributi che assegnano del codice eseguibile al gestore degli eventi di un controllo di interfaccia utente
 - `onclick`
 - `onmouseover`

Conclusioni

- Creare la propria pagina personale in HTML5
- La pagina deve avere le sezioni
 - Home
 - Vita
 - Attività
 - Hobby
 - Contatti
- Utilizzare almeno gli elementi HTML5 descritti in questa presentazione

Riferimenti



<https://w3schools.com>