

HTML

▼ Heading

I tag <h1>, <h2>, ..., <h6> servono per definire dei titoli di importanza decrescente. La h sta proprio per “heading”, cioè titolo. Sono elementi di blocco, lasciando infatti una linea vuota sia prima che dopo di essi.

▼ Paragrafi

Il paragrafo è l'unità base entro cui suddividere un testo: è un elemento tipico del blocco. Il tag <p> lascia una riga vuota prima e dopo la chiusura. Se si vuole andare a capo si usa l'elemento

▼ Div

E' l'elemento di tipo blocco per eccellenza: a differenza di <p> , <div> NON lascia spazi prima e dopo la sua apertura.

All'interno di questi blocchi, solitamente è annidato uno “span”, cioè un contenitore generico. E' un elemento inline e quindi NON va a capo ma continua sulla stessa linea del tag che lo include. E' un elemento molto usato soprattutto insieme ai fogli di stile per dare un aspetto particolare ad un pezzo di testo in un blocco.

▼ Horizontal rule

Serve per inserire una riga di separazione

▼ Liste non ordinate

Il tag (*unordered list*) permette di definire liste NON ordinate. Gli elementi della lista vengono definiti mediante il tag (*list item*). L'attributo type definisce la forma dei punti e ammette tre valori: disc, circle, square.

```
<ul type="disc">
  <li>info</li>
  <li>info</li>
  <li>info</li>
</ul>
```

▼ Liste ordinate

Il tag `` (*ordered list*) permette di definire liste ordinate. L'attributo `type` definisce il tipo di numerazione e ammette 5 valori: 1, a, A, i, I.

```
<ol type="I">
  <li>info</li>
  <li>info</li>
  <li>info</li>
</ol>
```

▼ Tabelle

Con il tag `<table>` creiamo una tabella. L'attributo `border` specifica la dimensione dei bordi.

`<tr>` è il tag che ci permette di definire il contenuto di una riga. `<th>` invece

▼ Form

Un **form** è una sezione di documento HTML che contiene elementi di controllo che l'utente può utilizzare per inserire dati o in generale per interagire. I dati inseriti possono essere inoltrati al server dove un agente può processarli. Gli elementi di controllo sono caratterizzati da un *valore iniziale* e da un *valore corrente*. Essi sono:

- **Bottoni**
- **Checkbox**
- **Radio Button**
- **Liste**
- **Caselle di inserimento testo**
- **Oggetti nascosti**

```
<form>
  <label for="name">Nome:</label>
  <input type="text" id="name" name="name">
  <input type="submit" value="Invia">
</form>
```

All'interno del tag `<form>` troviamo tutti i vari attributi:

- `action = uri` URI dell'agente che riceverà i dati del form
- `name = text` specifica il nome del form
- `method = {get | post}` specifica il modo in cui i dati vengono inviati
- `enctype = content-type` se il metodo è POST specifica il content type usato per la codifica dei dati contenuti nel form

La maggior parte dei controlli viene definita mediante il tag `<input>`. L'attributo **type** stabilisce il tipo di controllo:

▼ `text`

E' un campo di inserimento di testo su una sola riga.

```
<p> Nome: <input type="text" name="firstname" size=15 maxl
<!-- dove value è un eventuale valore iniziale, size lung
maxlength massima lunghezza del testo -->
```

▼ `password`

Come text ma non è leggibile (***)

▼ `file`

Consente di fare l'upload di un file selezionandolo dal filesystem del client.
richiede una codifica particolare per il form: `enctype = "multipart/form-data"`.

```
<p> <input type="file" name="attach">
```

▼ `checkbox`

Definisce una casella di spunta:

```
<p>
  <input type="checkbox" name="food" value="pane"> Pane
  <input type="checkbox" name="food" value="burro"> Burro
  <input type="checkbox" name="drink" value="acqua" checked=""
</p>
```

```
<!-- dove value è il valore restituito se la casella viene  
spuntiamo per default una casella -->
```

▼ radio button

E' come un checkbox, ma i Radio Button sono mutualmente esclusivi. O meglio, tutti i controlli che condividono lo stesso nome sono esclusivi tra loro. Anche qui possiamo utilizzare `checked`

```
<p>
    <input type="radio" name="sex" value="M"> Maschio <br>
    <input type="radio" name="sex" value="F"> Femmina <br/>
</p>
```

▼ button

Per i bottoni si utilizzano tre valori dell'attributo type:

- **submit:** per il bottone che provoca la spedizione del form al server
- **reset:** per il bottone che riporta il contenuto dei campi al valore originale
- **button:** per un generico bottone di azione

L'etichetta che compare nel bottone viene definita usando l'attributo `value`

[illegible]

▼ Liste di opzioni

Il tag `<select>` permette di costruire liste di opzioni. L'aspetto di default è quello di una combobox:

```
<select name="provincia" >
  <option value="BO" selected="selected">Bologna</option>
  <option value="MO">Modena</option>
```

```

<option value="RE">Reggio Emilia</option>
<option value="PR">Parma</option>
<option value="PC">Piacenza</option>
</select>
<!-- l'attributo selected="selected" indica una scelta predefinita

```

Se si utilizza l'attributo **multiple** NON abbiamo più una combo ma una lista aperta, nel senso che possiamo scegliere più opzioni.

Con il tag `optgroup` invece, è possibile organizzare la lista in gruppi

```

<select name="provincia" multiple="multiple" size=7>
  <optgroup label="Capoluogo">
    <option value="BO" selected="selected">Bologna</option>
  </optgroup>
  <optgroup label="Emilia">
    <option value="MO">Modena</option>
    <option value="RE">Reggio Emilia</option>
    <option value="PR">Parma</option>
    <option value="PC">Piacenza</option>
  </optgroup>
</select>

```

▼ cornice

Il tag `fieldset` creiamo una sorta di cornice nella pagina. Possiamo “scrivere sulla cornice” con il tag `legend`:

```

<fieldset style="border:solid">
  <legend> Questionario: </legend>
</fieldset>

```

└─ Questionario: _____

