

Corso di Web Programming

3. HTML Parte I

Paolo Milazzo

Dipartimento di Informatica, Università di Pisa

<http://www.di.unipi.it/~milazzo>

milazzo@di.unipi.it

Corso di Laurea in Informatica Applicata

A.A. 2010/2011

Sommario

- 1 Introduzione a HTML
- 2 Struttura di un documento HTML
- 3 Testo semplice e strutturato
 - Testo semplice
 - Liste
 - Tabelle
- 4 Immagini
- 5 I collegamenti ipertestuali
- 6 Le immagini-mappa

Introduzione a HTML (1)

- HTML sta per HyperText Markup Language
- HTML non è un linguaggio di programmazione, ma un linguaggio di markup costituito da un insieme di **tag**:
`<html>`, `<p>`, `<table>`, ``,
- Un documento HTML **descrive** una pagina web usando una opportuna combinazione di tag e testo semplice
- Il compito di un web browser è di interpretare un documento HTML e visualizzare la pagina web da esso descritta
- La versione più utilizzata di HTML è al momento la 4 (o meglio, 4.01)
- Recentemente è stata definita la versione 5

Introduzione a HTML (2)

Un documento HTML è essenzialmente un documento di testo arricchito con tag

Un frammento di HTML:

```
<h3>Formal Biological Modelling:</h3>
<ul>
<li> Synthetic biology, circuits design (IGEM models); </li>
<li> Formal methods for the representation of biological
systems (rewrite systems, process calculi, graph grammars,
hybrid systems, etc.);</li>
<li> Theoretical links and comparisons between different
formal models for the modelling of biological processes;</li>
<li> Quantitative (probabilistic, timed, stochastic, etc.)
languages and calculi;</li>
<li> Spatial languages and calculi.</li>
</ul>
```

Introduzione a HTML (3)

- Un tag normalmente prevede un'apertura (e.g. `<p>`) e una chiusura (e.g. `</p>`) e descrive il proprio contenuto:
 - ▶ Ad esempio, il tag `<p>` può essere usato così `<p>Hello world</p>` per specificare che la frase Hello world è un paragrafo
- Un tag può avere attributi
 - ▶ Ad esempio, il tag `<a>` può essere usato così `clicca qui` per specificare che "clicca qui" è un link al sito di Repubblica, indicato tramite l'attributo `href`
- Un tag che non ha contenuto può essere contratto in un unico simbolo
 - ▶ Il tag `<hr>` da solo crea una riga orizzontale lungo tutta la pagina (per definizione `<hr>` non può avere contenuto)
 - ▶ Tag che possono avere un contenuto (e.g. `<p>`) se usati senza contenuto devono essere scritti in uno dei seguenti modi:
 - ★ `<p></p>`
 - ★ `<p/>`
 - ▶ Anche per `<hr>` si può usare la sintassi `<hr/>`

Introduzione a HTML (4)

Qualche nota sintattica:

- Si possono inserire **commenti** (ignorati dal browser) con la seguente sintassi:
`<!-- Questo e' un commento -->`
- I tag html **non sono case sensitive**
- HTML collassa tutti i caratteri di whitespace (SPACE, TAB, CR, LF) in un unico spazio. Questo permette di organizzare il sorgente in maniera leggibile senza influenzare la visualizzazione sul browser
- Se nel testo di un documento HTML si vogliono usare lettere accentate e altri simboli riservati (e.g. < e >) bisogna usare le entità carattere:

à	à	è	è	é	é	ì	ì
ò	ò	ù	ù	À	À	È	È
É	É	Ì	Ì	Ò	Ò	Ù	Ù
>	>	<	<	&	&	"	"

Introduzione a HTML (5)

- Generalmente i browser web sono molto (anche troppo) tolleranti: accettano anche documenti HTML con errori cercando di visualizzare il più possibile
- Questo, e il fatto che nel corso del tempo i diversi browser hanno supportato in maniera diversa i vari tag HTML ha fatto sì che il linguaggio non sia mai riuscito a convergere veramente in un'unica versione
- Più recentemente è stato definito anche XHTML, una versione di HTML come dialetto XML che impone vincoli sintattici più forti
 - ▶ tag scritti in minuscolo
 - ▶ obbligo di avere sempre il tag di chiusura
 - ▶ obbligo di usare le virgolette negli attributi
 - ▶

Introduzione a HTML (6)

- Un buon uso di HTML prevede la separazione tra la descrizione della struttura della pagina web e la descrizione degli aspetti di presentazione
 - ▶ La **struttura** di una pagina web include la presenza di titoli, liste, tabelle, immagini, link, ecc...
 - ▶ Gli aspetti di **presentazione** invece includono i font e i colori utilizzati, il posizionamento dei contenuti nella pagina, ecc....
- Sebbene includa alcuni aspetti presentazionali, HTML dovrebbe essere usato solo per descrivere la struttura della pagina
- Per descrivere gli aspetti di presentazione è bene usare il linguaggio CSS (Cascading Style Sheets), che vedremo in seguito

Struttura di un documento HTML (1)

Un documento HTML contiene due parti:

- L'header, che contiene solo informazioni sul documento stesso;
- Il corpo (body), che è la parte di documento che viene effettivamente visualizzata.

L'intero documento inizia con il tag `<html>` (chiuso da `</html>`), mentre le due parti header e corpo sono delimitate dai tag `<head>` e `<body>`, rispettivamente.

```
<html>
  <head>
    ..... header del documento .....
  </head>
  <body>
    ..... corpo del documento .....
  </body>
</html>
```

L'header di un documento HTML (1)

All'interno dell'header di un documento html è comune trovare:

- Il titolo del documento (che apparirà in cima alla finestra del browser) usando il tag `<title>`. Ad esempio:
 - ▶ `<title>Corso di Web Programming</title>`
- La definizione di una base per gli URL relativi usati nella pagina usando il tag `<base>`. Ad esempio:
 - ▶ `<base href="http://www.di.unipi.it/~milazzo">`
- Meta dati (informazioni aggiuntive) sul documento usate dalle applicazioni e dai motori di ricerca. Si inseriscono usando il tag `<meta>`.

L'header di un documento HTML (2)

Il tag `<meta>` ha il seguente formato:

```
<meta http-equiv="xxx" name="yyy" content="zzz" scheme="www">
```

- l'attributo `http-equiv` serve per trasmettere informazioni che si vorrebbero aggiungere all'header della risposta http con cui viene trasmesso il documento.
 - ▶ Esempio classico: ridirezione automatica di una pagina che ha cambiato indirizzo:
 - ▶ `<meta http-equiv="refresh" content="5; url=http://www.nuovosito.it/index.html"/>`
- l'attributo `name` specifica il tipo di informazione che si vuole descrivere;
- l'attributo `content` specifica il contenuto dell'informazione stessa;
- l'attributo `scheme` è raramente usato (serve in alcuni casi per disambiguare l'informazione).

L'header di un documento HTML (3)

Gli utilizzi più comuni del tag `<meta>` (corrispondenti a diversi valori per l'attributo `name`) sono i seguenti:

- Specificare l'autore del documento
 - ▶ `<meta name="author" content="Paolo Milazzo">`
- Dare un breve descrizione del contenuto del documento
 - ▶ `<meta name="description" content="Sito ufficiale del corso di Laurea su Web Programming">`
- Elencare parole chiave che rappresentano il documento
 - ▶ `<meta name="keywords" content="html php www http mysql laurea informatica paolo milazzo">`

L'header di un documento HTML (4)

Queste informazioni sono usate dai motori di ricerca per indicizzare la pagina.

- Un buon uso (in particolare delle keywords) dare maggiore visibilità alla pagina;
- Attenzione a non fare abusare di keywords ricercatissime (e.g. sesso, mp3,) solo per migliorare il posizionamento della propria pagina nella lista dei motori di ricerca. Quando i motori di ricerca se ne accorgono, rimuovono la pagina dalle loro liste.

L'header di un documento HTML (5)

A questo punto un esempio di bozza di documento HTML dotato di header è il seguente:

```
<html>
  <head>
    <title>Pagina di prova</title>
    <meta name="author" content="Paolo Milazzo">
  </head>
  <body>
    ..... corpo del documento .....
  </body>
</html>
```

Il corpo di un documento HTML (1)

Il corpo di un documento HTML può contenere un sacco di cose:

- Testo (semplice e strutturato)
- Liste (numerata e non numerata)
- Tabelle
- Collegamenti ipertestuali (link ad altri documenti o a altre parti dello stesso documento)
- Immagini
- Mappe (immagini cliccabili)
- Frames
- Form (moduli compilabili dall'utente)
- Script (frammenti di codice eseguibile)

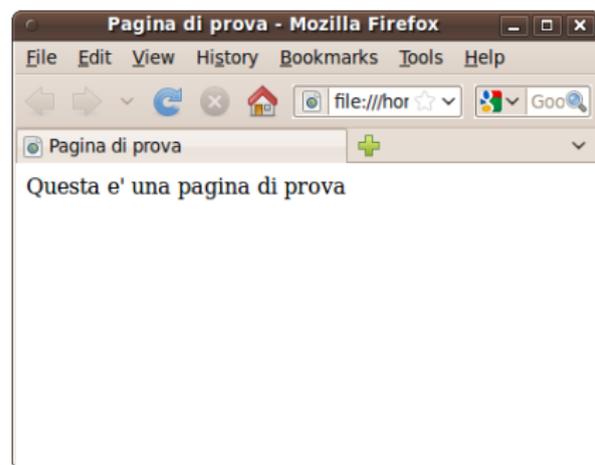
Il corpo di un documento HTML (2)

La forma di contenuto più immediata è il testo semplice.

- Sebbene possa essere inserito senza usare tag particolari è bene racchiudere i vari paragrafi di testo nel tag `<p>`.

Inserendo un po' di testo nel corpo del nostro esempio otteniamo:

```
<html>
  <head>
    <title>Pagina di prova</title>
    <meta name="author"
          content="Paolo Milazzo">
  </head>
  <body>
    <p>Questa e' una pagina
      di prova</p>
  </body>
</html>
```



Il corpo di un documento HTML (3)

Vediamo ora uno per uno tutti i tipi di contenuti che possono essere usati nel corpo di un documento HTML.

Negli esempi che faremo ometteremo la struttura dell'intero documento, ma illustreremo solo il frammento di codice HTML di interesse.

Si assume quindi che i frammenti di codice HTML illustrati saranno da inserire all'interno della parte `<body> . . . </body>` di un documento HTML completo (e.g. l'esempio che abbiamo già visto).

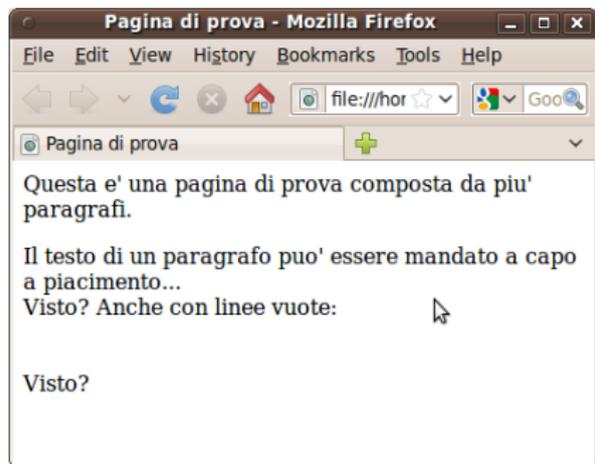
Il testo semplice: paragrafi

Il modo più semplice per inserire testo in una pagina HTML è tramite paragrafi:

- Si usa un tag `<p>` per ogni paragrafo;
- I paragrafi sono “unità logiche” di testo, spesso separati nella visualizzazione tramite spazio verticale o indentazione della prima riga;
- Se è necessario inserire un'interruzione di linea all'interno di un paragrafo basta inserire il tag `
` (usato più volte crea righe vuote)

```
<p>Questa e' una pagina di prova  
composta da piu' paragrafi.</p>
```

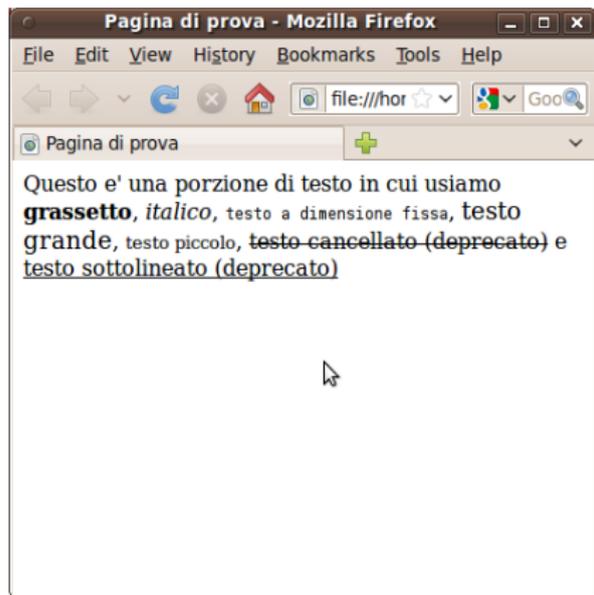
```
<p>Il testo di un paragrafo  
puo' essere mandato a capo  
a piacimento...<br>  
Visto? Anche con linee vuote:  
<br><br><br>  
Visto?  
</p>
```



Il testo semplice: stili (1)

- HTML prevede alcuni stili (grassetto, italico, ecc...) che possono essere assegnati a porzioni di testo, come nei seguenti esempi.
- L'uso di alcuni di questi stili è deprecato (si consiglia di usare fogli di stile)

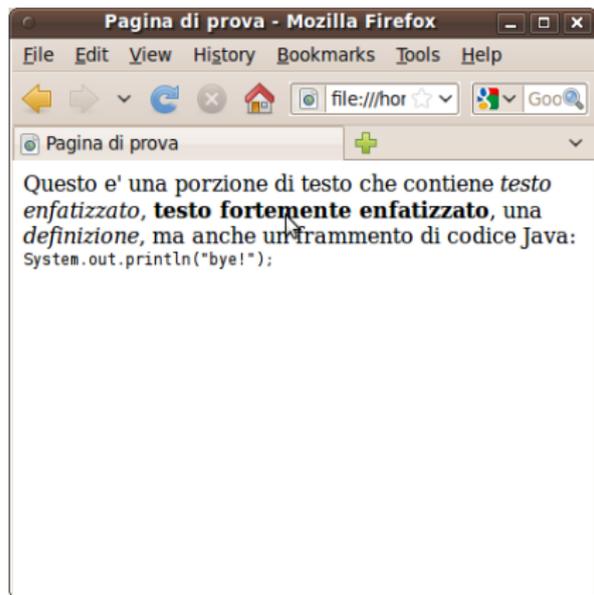
```
<p> Questo e' una porzione di
testo in cui usiamo
<b>grassetto</b>, <i>italico</i>,
<tt>testo a dimensione fissa</tt>,
<big>testo grande</big>,
<small> testo piccolo</small>,
<s>testo cancellato
(deprecato)</s> e <u>testo
sottolineato (deprecato)</u>
</p>
```



Il testo semplice: stili (2)

- HTML prevede anche alcuni stili legati all'interpretazione logica del testo (testo enfatizzato, definizioni,).
- La visualizzazione (dimensioni del font, ecc...) può dipendere dal browser usato.

```
<p> Questo e' una porzione di
testo che contiene
<em>testo enfatizzato</em>,
<strong>testo fortemente
enfatizzato</strong>, una
<dfn>definizione</dfn>, ma anche
un frammento di codice Java:
<code>System.out.println("bye!");
</code></p>
```



Il testo semplice: testo preformattato

- E' inoltre possibile inserire in un documento HTML del testo preformattato, ovvero in cui gli spazi bianchi e le interruzioni di linea hanno un significato (e.g. in una poesia);
- Il browser spesso usa un font diverso per il testo preformattato

```
<p>E ora un po' di testo
preformattato:
<pre>
La formattazione di
questo testo non viene modificata
                nella visualizzazione
</pre>
</p>
```



Il testo semplice: intestazioni

- Il testo in un documento può essere usato come intestazione o titolo (dell'intero documento, di una sezione, ecc...);
- HTML prevede 6 livelli di importanza per le intestazioni, corrispondenti ai tag <h1>,<h2>,...,<h6>;
- La visualizzazione (dimensioni del font, ecc...) può dipendere dal browser usato.

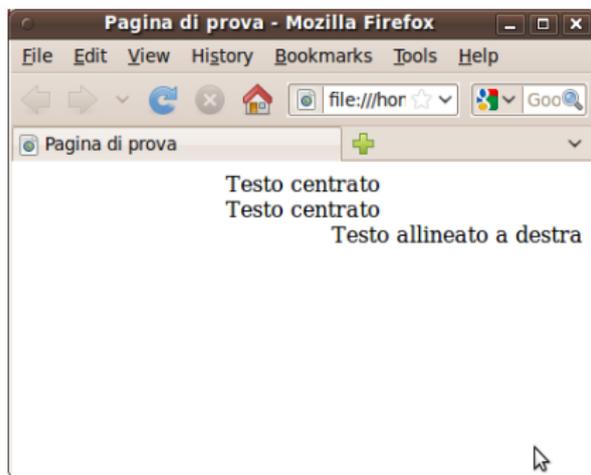
```
<h1>Titolo con h1</h1>  
<h2>Titolo con h2</h2>  
<h3>Titolo con h3</h3>  
<h4>Titolo con h4</h4>  
<h5>Titolo con h5</h5>  
<h6>Titolo con h6</h6>  
<p>Testo semplice</p>
```



Il testo semplice: allineamento

- Spesso è preferibile centrare o allineare il testo (e non solo) a destra;
- Tradizionalmente si può centrare qualunque cosa usando `<center>`;
- Alternativamente si può usare il tag `<div align=xxx>` con `xxx` che può essere `left`, `right`, `center` o `justify` ;
 - ▶ Il tag `<div>` in generale serve per raggruppare una porzione del documento a cui applicare un'unica formattazione (vedremo...);
- E' bene dire che questi metodi sono però deprecati, e che l'allineamento del testo dovrebbe essere fatto usando fogli di stile.

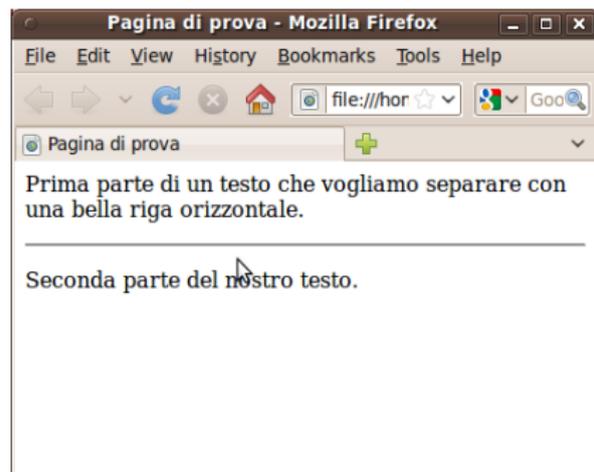
```
<p>
<center>Testo centrato</center>
<div align=center>
Testo centrato</div>
<div align=right>
Testo allineato a destra</div>
</p>
```



Il testo semplice: linee di separazione

- Spesso è comodo inserire linee orizzontali per separare porzioni di testo semplicemente usando `<hr>`;
- Questo tag crea una linea orizzontale larga quanto l'intera finestra del browser.
 - ▶ si ridimensiona automaticamente ridimensionando la finestra

```
<p>Prima parte di un testo che  
vogliamo separare con una  
bella riga orizzontale.</p>  
<hr>  
<p>Seconda parte del nostro testo.  
</p>
```



Il testo semplice: un esempio completo (1)

```
<html>
<head><title>Manuale per l'utente</title></head>
<body>

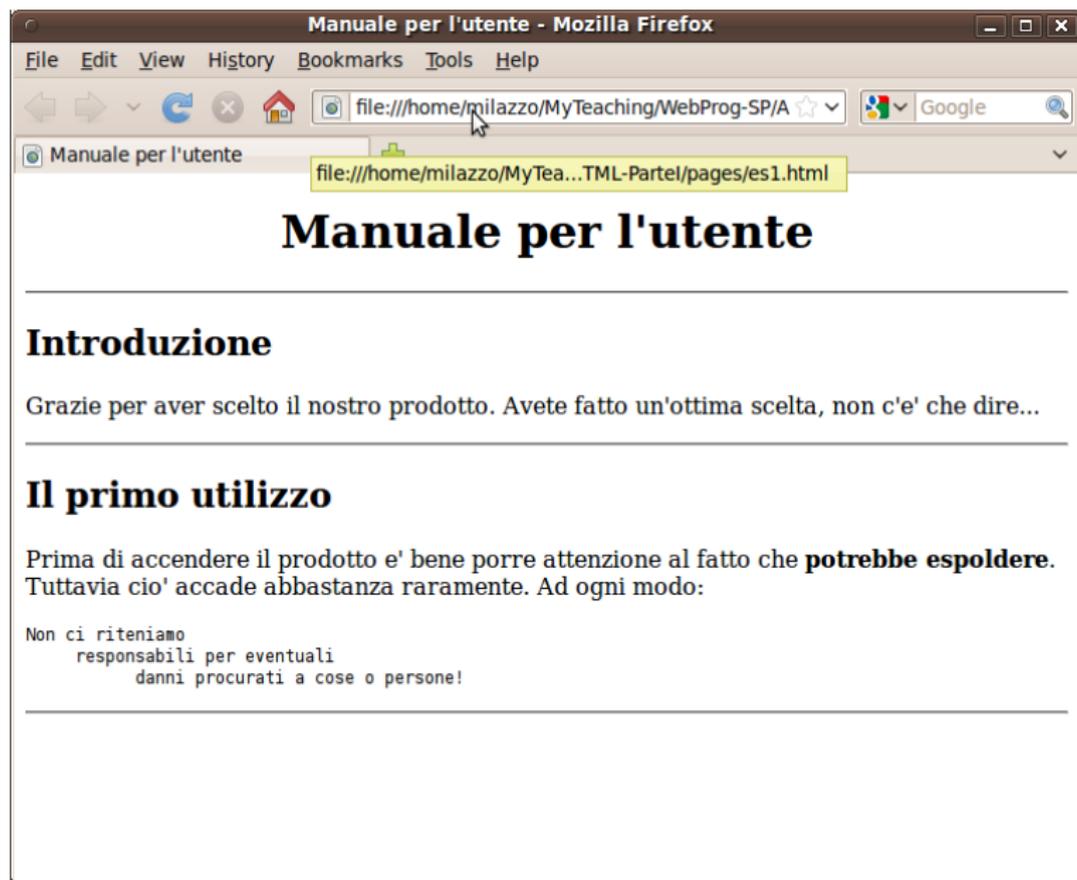
<center><h1>Manuale per l'utente</h1></center>
<hr>

<h2>Introduzione</h2>
<p>Grazie per aver scelto il nostro prodotto. Avete fatto
un'ottima scelta, non c'e' che dire...</p>
<hr>

<h2>Il primo utilizzo</h2>
<p>Prima di accendere il prodotto e' bene porre attenzione
al fatto che <strong>potrebbe espordere</strong>. Tuttavia cio'
accade abbastanza raramente. Ad ogni modo:
<pre>
Non ci riteniamo
    responsabili per eventuali
        danni procurati a cose o persone!
</pre>
</p>
<hr>

</body>
</html>
```

Il testo semplice: un esempio completo (2)



Manuale per l'utente - Mozilla Firefox

File Edit View History Bookmarks Tools Help

file:///home/milazzo/MyTeaching/WebProg-SP/A

Manuale per l'utente

file:///home/milazzo/MyTea...TML-Partel/pages/es1.html

Manuale per l'utente

Introduzione

Grazie per aver scelto il nostro prodotto. Avete fatto un'ottima scelta, non c'e' che dire...

Il primo utilizzo

Prima di accendere il prodotto e' bene porre attenzione al fatto che **potrebbe espordere**. Tuttavia cio' accade abbastanza raramente. Ad ogni modo:

Non ci riteniamo
responsabili per eventuali
danni procurati a cose o persone!

Le liste (1)

HTML prevede tre tipi di liste:

- liste non ordinate (unordered lists)
- liste numerate (ordered lists)
- liste di definizioni (definition lists)

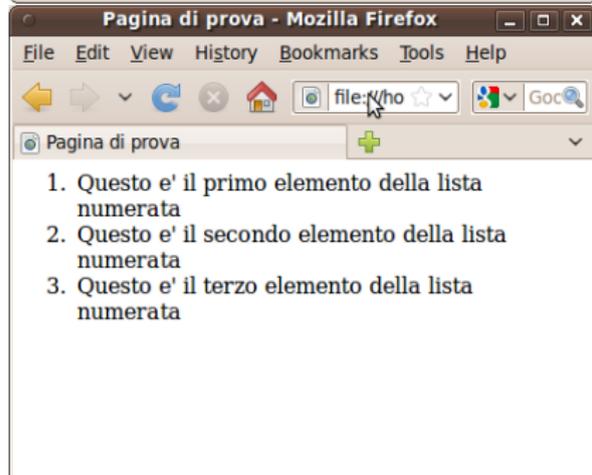
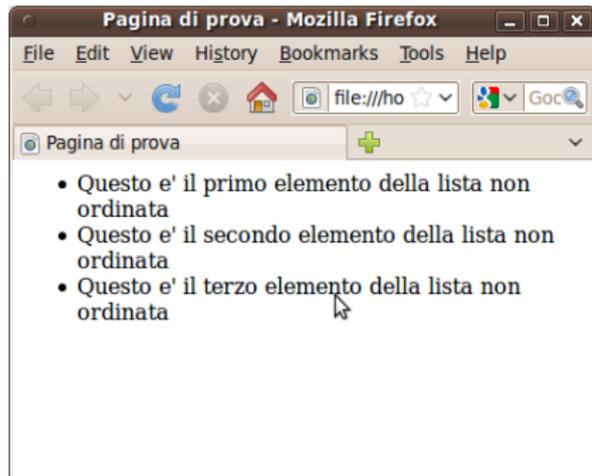
La sintassi delle liste non ordinate e numerate è simile:

- L'intera lista è racchiusa in `` (per liste non ordinate) o in `` (per liste numerate);
- Ogni elemento è racchiuso in `` (per entrambi i tipi di lista) – `` può essere omesso.

Le liste (2)

```
<ul>
<li>Questo e' il primo elemento
della lista non ordinata
<li>Questo e' il secondo elemento
della lista non ordinata
<li>Questo e' il terzo elemento
della lista non ordinata
</ul>
```

```
<ol>
<li>Questo e' il primo elemento
della lista numerata
<li>Questo e' il secondo elemento
della lista numerata
<li>Questo e' il terzo elemento
della lista numerata
</ol>
```

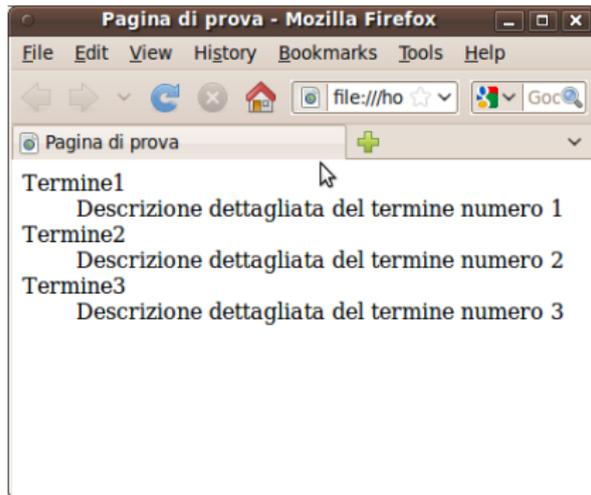


Le liste (3)

La sintassi delle liste di definizioni è la seguente:

- L'intera lista è racchiusa in `<dl>`;
- Ogni elemento è composto da due parti:
 - ▶ Il termine definito per il quale si usa il tag `<dt>`
 - ▶ La descrizione del termine per il quale si usa il tag `<dd>`
 - ▶ La chiusura di `<dt>` e `<dd>` è opzionale

```
<dl>
<dt>Termine1
<dd>Descrizione dettagliata del
termine numero 1
<dt>Termine2
<dd>Descrizione dettagliata del
termine numero 2
<dt>Termine3
<dd>Descrizione dettagliata del
termine numero 3
</dl>
```



Le tabelle (1)

- Nei documenti HTML si possono inserire dati che hanno una naturale rappresentazione come tabelle tramite il tag `<table>`;
- Le tabelle vengono spesso utilizzate anche come strumento di organizzazione della visualizzazione (layout) dei contenuti della pagina;
 - ▶ per questo il tag `<table>` (e i tag relati) prevedono un sacco di opzioni di formattazione;
 - ▶ L'uso di tabelle come strumento di layout è però deprecato (si dovrebbero invece usare fogli di stile)

Le tabelle (1)

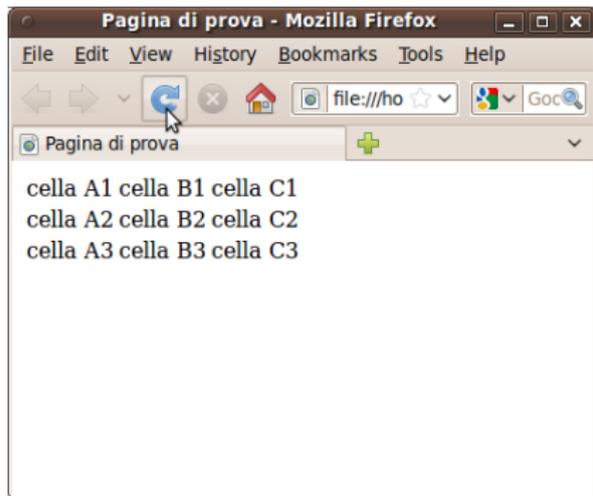
La struttura di base di una tabella è la seguente:

- L'intera tabella è racchiusa nel tag `<table>` che contiene:
- una lista di righe, ogniuna racchiusa nel tag `<tr>`, che a sua volta contiene:
- una lista di elementi (uno per ogni colonna) racchiusi nel tag `<td>`

```
<table>
<tr>
<td>cella A1</td>
<td>cella B1</td>
<td>cella C1</td>
</tr>

<tr>
<td>cella A2</td>
<td>cella B2</td>
<td>cella C2</td>
</tr>

<tr>
<td>cella A3</td>
<td>cella B3</td>
<td>cella C3</td>
</tr>
</table>
```



Le tabelle (2)

Alcune celle possono anche rappresentare titoli di righe o colonne:

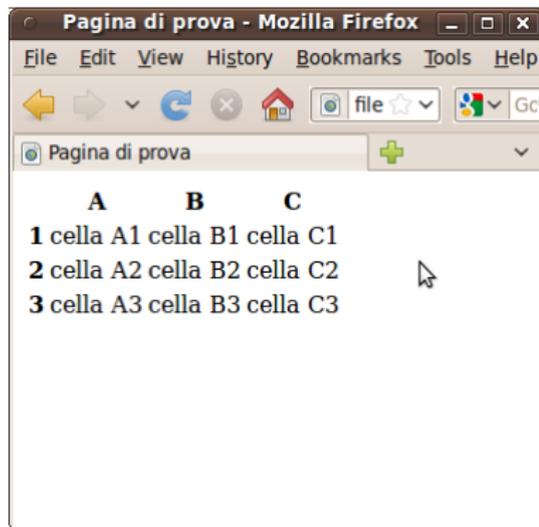
- per queste si usa `<th>` (table head) al posto di `<td>` (table data)
- la formattazione di queste celle è solitamente diversa dalle altre (a discrezione del browser)

```
<table>
<tr>
<th><!--cella vuota in angolo--></th>
<th>A</th> <th>B</th> <th>C</th>
</tr>

<tr>
<th>1</th>
<td>cella A1</td> <td>cella B1</td>
<td>cella C1</td>
</tr>

.....

<tr>
<th>3</th>
<td>cella A3</td> <td>cella B3</td>
<td>cella C3</td>
</tr>
</table>
```



Le tabelle (3)

E' anche possibile specificare celle che spaziano su più righe o colonne:

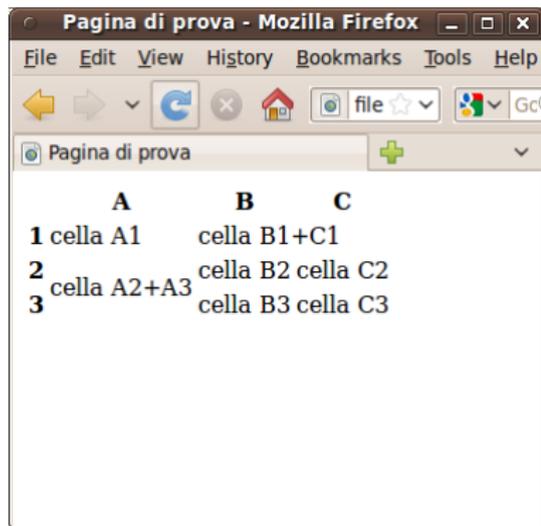
- usando gli attributi `rowspan` e `colspan` del tag `<td>`
- il valore degli attributi è il numero di righe o colonne su cui spaziare

```
<table>
<tr>
<th><!--cella vuota in angolo--></th>
<th>A</th> <th>B</th> <th>C</th>
</tr>

<tr>
<th>1</th>
<td>cella A1</td>
<td colspan=2>cella B1+C1</td>
</tr>

<tr>
<th>2</th>
<td rowspan=2>cella A2+A3</td>
<td>cella B2</td> <td>cella C2</td>
</tr>

<tr>
<th>3</th>
<td>cella B3</td> <td>cella C3</td>
</tr>
</table>
```



Le tabelle: aspetti di formattazione (1)

- Vediamo ora alcuni aspetti di formattazione delle tabelle;
- Ma ricordiamo che nei siti web “professionali” gli aspetti di formattazione dovrebbero essere trattati in un foglio di stile separato

Le tabelle: aspetti di formattazione (2)

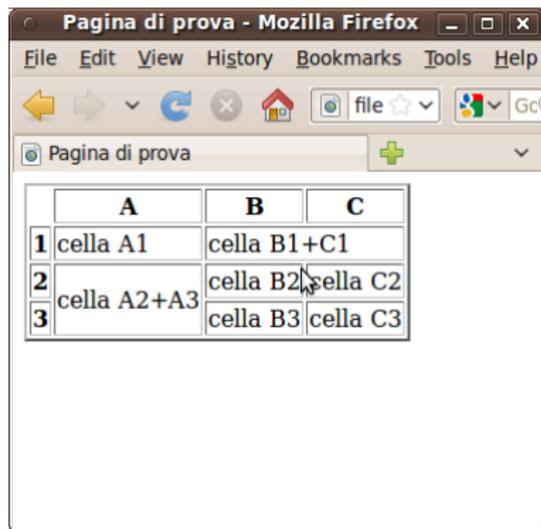
- E' possibile definire il bordo di una tabella tramite l'attributo border del tag <table>;
- Il valore dell'attributo è lo spessore del bordo espresso in pixel

```
<table border=2>
<tr>
<th><!--cella vuota in angolo--></th>
<th>A</th> <th>B</th> <th>C</th>
</tr>

<tr>
<th>1</th>
<td>cella A1</td>
<td colspan=2>cella B1+C1</td>
</tr>

<tr>
<th>2</th>
<td rowspan=2>cella A2+A3</td>
<td>cella B2</td> <td>cella C2</td>
</tr>

<tr>
<th>3</th>
<td>cella B3</td> <td>cella C3</td>
</tr>
</table>
```



Le tabelle: aspetti di formattazione (3)

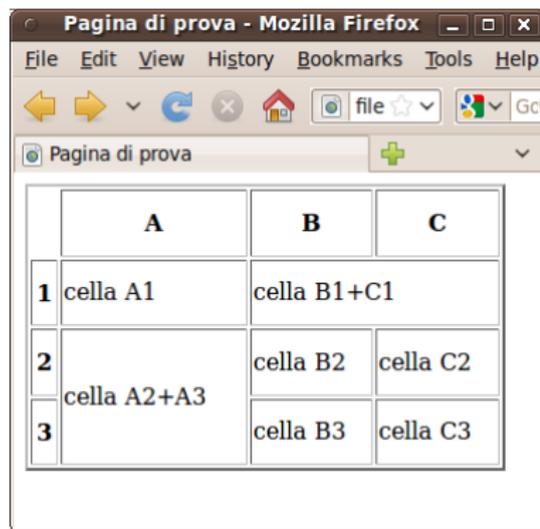
- Gli attributi `width` e `height` specificano larghezza e altezza della tabella (in pixel o in % sulle misure della finestra del browser)
 - ▶ Con un valore percentuale la tabella si ridimensiona automaticamente quando si ridimensiona la finestra del browser

```
<table border=2 width=95% height=200>
<tr>
<th><!--cella vuota in angolo--></th>
<th>A</th> <th>B</th> <th>C</th>
</tr>

<tr>
<th>1</th>
<td>cella A1</td>
<td colspan=2>cella B1+C1</td>
</tr>

<tr>
<th>2</th>
<td rowspan=2>cella A2+A3</td>
<td>cella B2</td> <td>cella C2</td>
</tr>

<tr>
<th>3</th>
<td>cella B3</td> <td>cella C3</td>
</tr>
</table>
```



Le tabelle: aspetti di formattazione (4)

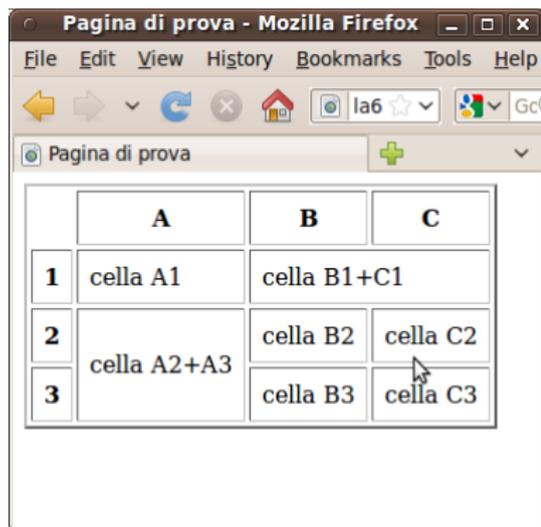
- L'attributo `cellspacing` specifica lo spazio tra le celle (in pixel)
- L'attributo `cellpadding` specifica lo spazio vuoto tra il bordo di una cella e il suo contenuto (in pixel)

```
<table border=2 cellspacing=3 cellpadding=8>
<tr>
<th><!--cella vuota in angolo--></th>
<th>A</th> <th>B</th> <th>C</th>
</tr>

<tr>
<th>1</th>
<td>cella A1</td>
<td colspan=2>cella B1+C1</td>
</tr>

<tr>
<th>2</th>
<td rowspan=2>cella A2+A3</td>
<td>cella B2</td> <td>cella C2</td>
</tr>

<tr>
<th>3</th>
<td>cella B3</td> <td>cella C3</td>
</tr>
</table>
```



Le tabelle: aspetti di formattazione (5)

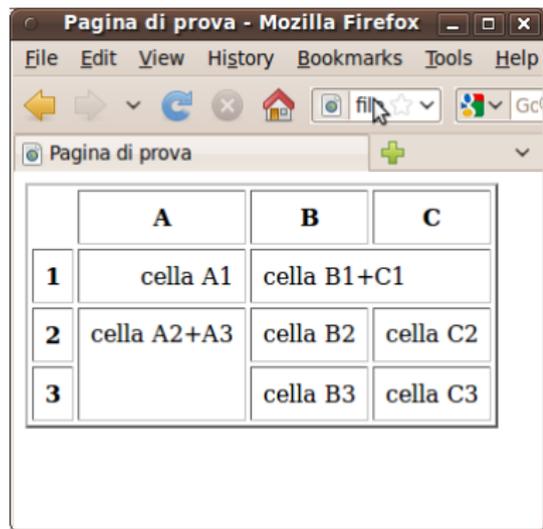
- Gli attributi `align` e `valign` possono essere usati in `<table>`, `<tr>` e `<td>` per specificare l'allineamento orizzontale e verticale del testo nella tabella, in una riga o in una cella;
 - ▶ Valori per `align`: `left`, `right`, `center`;
 - ▶ Valori per `valign`: `top`, `middle`, `bottom`;

```
<table border=2 cellspacing=3 cellpadding=8>
<tr>
<th><!--cella vuota in angolo--></th>
<th>A</th> <th>B</th> <th>C</th>
</tr>
```

```
<tr>
<th>1</th>
<td align="right">cella A1</td>
<td colspan=2>cella B1+C1</td>
</tr>
```

```
<tr align="center" valign="top">
<th>2</th>
<td rowspan=2>cella A2+A3</td>
<td>cella B2</td> <td>cella C2</td>
</tr>
```

```
<tr>
<th>3</th>
<td>cella B3</td> <td>cella C3</td>
</tr>
```



Le immagini (1)

- Un'immagine può essere inserita tramite il tag ``
- Il tag `` prevede solo attributi (non un contenuto) ed ha il formato `` dove:
 - ▶ `url` è un riferimento all'immagine che si vuole inserire
 - ▶ `xxxx` è una breve descrizione dell'immagine che viene visualizzata (di solito) al posto dell'immagine in alcuni casi (url sbagliata o immagini disattivate nel browser) o quando si passa con il mouse sull'immagine stessa.

```

```



Le immagini (2)

- E' anche possibile specificare le dimensioni di un'immagine (larghezza e altezza) usando gli attributi `width` e `height`
- Tali attributi prendono (di default) valori espressi in pixels come segue:
``.
- E' buona norma indicare le dimensioni di un'immagine in modo che il browser possa riservare lo spazio durante il caricamento della pagina (il layout della pagina non cambia durante il caricamento)

```

```



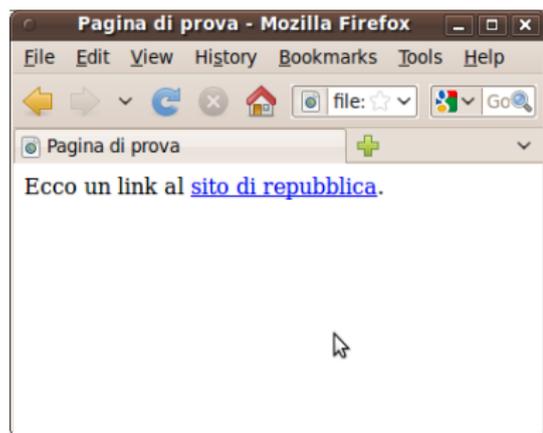
I collegamenti (link) ipertestuali (1)

- La caratteristica particolare degli ipertesti è di poter contenere collegamenti (link) tra le varie parti di un documento
- Il modo più semplice per realizzare collegamenti ipertestuali in HTML è tramite l'uso di "ancore"
- Le ancore possono essere usate in due modi:
 - ▶ per riferire a un documento HTML esterno, tramite la sua URL
 - ▶ per riferire a una parte diversa dello stesso documento HTML (e.g. a una sotto-sezione, all'inizio del documento, ecc...)
- Le ancore si definiscono usando il tag `<a>`

I collegamenti (link) ipertestuali (2)

- Per creare un link a un documento HTML esterno è sufficiente specificarne l'URL nell'attributo href del tag <a>
- Una volta visualizzato nel browser, tutto il contenuto del tag <a> diventerà "cliccabile", e solitamente il browser lo mostrerà usando uno stile diverso dal resto del documento (e.g. sottolineato e colorato – se testo)
- L'utente potrà seguire il link (ossia, scaricare il documento HTML dall'URL specificata) cliccando sul contenuto del tag <a>

```
Ecco un link al  
<a href="http://www.repubblica.it">  
sito di repubblica</a>.
```



I collegamenti (link) ipertestuali (3)

- Quando si vuole riferire a un documento diverso, ma dello stesso sito (nello stesso file system) si può usare un URL relativo
- Questo rende facile “spostare” un intero sito da una directory del file system ad un'altra
 - ▶ se si sposta l'intera directory contenente tutti di documenti HTML gli indirizzi relativi rimangono invariati

```
Ecco un esempio di
<a href="index.html">link a un documento nella stessa directory</a>
del documento corrente, un esempio di
<a href="subdir/pagina.html">link a un documento in
una sotto-directory</a> e un esempio di
<a href="../index.html">link a un documento in una
directory superiore</a>.
```

I collegamenti (link) ipertestuali (4)

- E' anche possibile creare link a parti diverse dello stesso documento HTML
 - ▶ e.g. a una sotto-sezione, all'inizio del documento, ecc...
- Per fare ciò è però necessario usare due volte il tag `<a>`
 - ▶ La prima volta dove si vuole che il link appaia (ancora sorgente)
 - ▶ La seconda volta nella parte del documento che deve essere raggiunta (ancora di destinazione)
- L'ancora di destinazione dovrà contenere un identificativo (una stringa) che sia univoca all'interno dell'intero documento (attributo `name` di `<a>`)
- L'ancora sorgente userà tale identificativo per riferire precisamente alla destinazione (attributo `href` di `<a>`)

I collegamenti (link) ipertestuali (5)

```
<a name = "inizio" />
<p>
Questa pagina e' composta di due parti:
la <a href = "#primaparte">prima parte</a>
e la <a href = "#secondaparte">seconda parte</a>
</p>

<hr>

<a name = "primaparte" />
<h3>Prima parte</h3>
<p>Questa e' la prima parte</p>

<p><a href = "#inizio">Torna all'inizio della pagina</a></p>

<hr>

<a name = "secondaparte" />
<h3>Seconda parte</h3>
<p>Questa e' la seconda parte</p>

<p><a href = "#inizio">Torna all'inizio della pagina</a></p>
```

(vedere l'esempio links-interni.html)

I collegamenti (link) ipertestuali (6)

Altre cose riguardo i collegamenti ipertestuali:

- Tutto ciò che si trova all'interno del tag `<a>` diventa cliccabile, non solo il testo:
 - ▶ anche immagini, intere tabelle, ecc...
- Si può anche specificare che il documento riferito da un link debba essere aperto in una nuova finestra del browser tramite l'attributo `target` di `<a>`
 - ▶ il valore da dare all'attributo è `_blank`
- Vedremo altre funzionalità e altri valori di questo attributo quando faremo i frames HTML

Le mappe (image maps) (1)

- Le mappe (image maps) sono immagini con aree cliccabili.
- Si definiscono usando i tag `` e `<map>` dove
 - ▶ `` richiama l'immagine
 - ▶ `<map>` definisce le aree cliccabili e i collegamenti corrispondenti
- Il tag `` si usa come al solito, ma con in più l'attributo `usemap` che specifica un identificativo di mappa (preceduto da #)
- Il tag `<map>` invece
 - ▶ ha un attributo `name` in cui si indica l'identificativo che lo lega all'immagine (senza #)
 - ▶ contiene una lista di aree (definite usando il tag `<area>`) che hanno una forma (rettangolo, cerchio o poligono), delle coordinate che ne definiscono posizione e dimensioni, un `href` (come per `<a>`) e un testo alternativo
- Le mappe sono usate spesso per creare indici di siti web

Le mappe (image maps) (2)

```


<map name="menumap">
  <area shape="rect" coords="7,16,203,117"
        href="home.html" alt="Home Page"/>
  <area shape="rect" coords="7,137,203,238"
        href="pictures.html" alt="Pictures"/>
  <area shape="rect" coords="8,250,204,351"
        href="links.html" alt="Links"/>
  <area shape="rect" coords="7,368,203,469"
        href="contacts.html" alt="Contacts"/>
</map>
```

Home

Pictures

Links

Contacts

Le mappe (image maps) (3)

- Le aree possono avere forma di rettangolo (`shape="rect"`), cerchio (`shape="circle"`) o poligono (`shape="poly"`)
- Inoltre `shape="default"` rappresenta l'intera immagine
- Le coordinate variano a seconda del valore dell'attributo `shape`
- Le mappe di solito non si fanno "a mano". Esistono tool e siti web che aiutano a realizzare mappe per una immagine data.