

Metodologie Informatiche Applicate al Turismo

8. 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 Scienze del Turismo

A.A. 2013/2014

Sommario

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

Introduzione a HTML (1)

- Una pagina web è, generalmente, un “documento” HTML con eventualmente annessi altri contenuti ausiliari
 - ▶ Immagini, audio, video
 - ▶ Altri elementi per la corretta visualizzazione del documento e per l'interazione con l'utente (CSS, JavaScript,)
- HTML sta per HyperText Markup Language
- HTML non è un linguaggio di programmazione, ma un linguaggio di **markup** (marcatura) costituito da un insieme di **tag**:
`<html>`, `<p>`, `<table>`, `<img>`,
 - ▶ Un documento HTML non rappresenta un programma da eseguire
 - ▶ Un documento HTML **describe** 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
 - ▶ L'operazione di visualizzare una pagina web si definisce **rendering** della pagina
- La versione più utilizzata di HTML è al momento la 4 (o meglio, 4.01)
- Recentemente è stata definita la versione 5, i principali browser sono già in grado di utilizzarla (almeno in parte)

Introduzione a HTML (2)

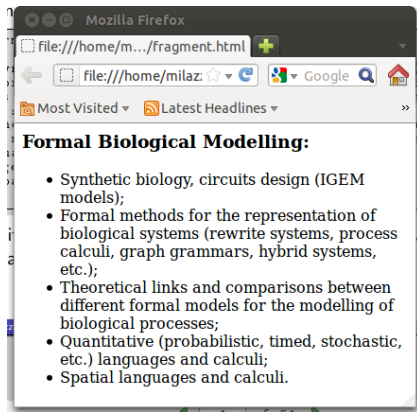
Un documento HTML è essenzialmente un documento di testo arricchito con tag

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)

Il compito di un browser è di prendere un documento HTML e “disegnarlo” sullo schermo



Introduzione a HTML (4)

- 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

Documento HTML

- Un documento HTML è un file di testo con una particolare struttura
- Un file HTML ha estensione **.html** o **.htm**
- I documenti HTML possono essere creati utilizzando un qualunque editor di testi (semplice programma di scrittura).
 - ▶ Il “blocco note” di Windows va benissimo per scrivere documenti HTML
- Esistono anche editor specifici per HTML, di due tipi:
 - ▶ Editor WYSIWYG (What You See Is What You Get)
 - ▶ Editor di testi specializzati

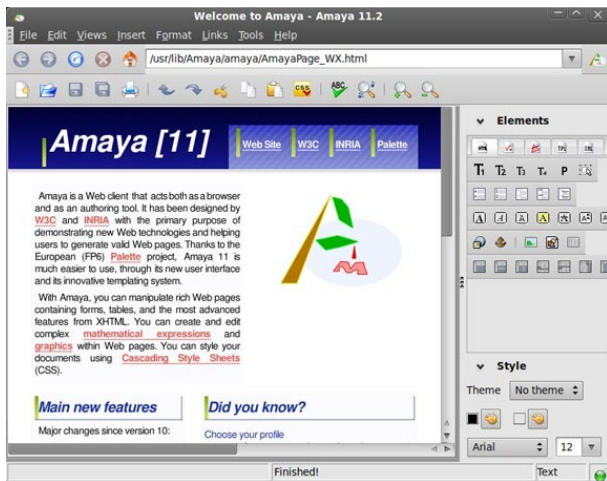
Editor WYSIWYG (1)

- Un editor WYSIWYG è un programma che consente di realizzare un documento HTML senza dover scrivere manualmente il codice HTML
- Funziona in maniera simile a Microsoft Word... si lavora direttamente sul documento come apparirà nel browser
 - ▶ Esempio: per scrivere un testo in grassetto si scrive il testo stesso e si clicca su un bottone che lo trasforma in grassetto
 - ▶ Il codice HTML viene **generato** (creato) dall'editor
- Vantaggi:
 - ▶ non richiede di conoscere il linguaggio HTML
 - ▶ consente di realizzare una pagina web rapidamente
- Svantaggi:
 - ▶ il codice HTML generato spesso non è di buona qualità (e.g. non funziona bene in tutti i browser)

Editor WYSIWYG (2)

- Esistono diversi editor WYSIWYG, sia open-source (quindi gratuiti) che commerciali, tra i quali:
 - ▶ Amaya (Open-source, <http://www.w3.org/Amaya/>)
 - ▶ CoffeeCup (Commerciale, <http://www.cooffeecup.com>)
 - ▶ Dreamweaver (Commerciale, <http://www.adobe.com/products/dreamweaver>)
 - ▶ Visual Web Developer (Gratuito, <http://www.microsoft.com/express/download>)

Editor WYSIWYG (3)



Amaya

Editor HTML testuali (1)

- Gli editor HTML testuali sono programmi che facilitano la scrittura del codice HTML
 - ▶ Controllano che il codice sia scritto correttamente
 - ▶ Facilitano la scrittura colorando le varie parti del codice con colori diversi
 - ▶ Includono dei menù (wizard) che consentono di generare frammenti di codice HTML da modificare successivamente a mano
- Vantaggi:
 - ▶ consentono di avere il massimo controllo sul documento realizzato
 - ▶ aiutano a prevenire gli errori nella scrittura del codice HTML
- Svantaggi:
 - ▶ richiedono di conoscere il linguaggio HTML
 - ▶ richiedono più tempo (rispetto agli editor WYSIWYG) per realizzare una pagina

Editor HTML testuali (2)

- Esistono tantissimi editor HTML testuali, sia gratuiti che commerciali, tra i quali:
 - ▶ Notepad++ (Gratuito)
 - ▶ HTMLKit (Sia gratuito che commerciale)
 - ▶ NoteTab (Sia gratuito che commerciale)
 - ▶

Editor HTML testuali (3)

[illegible]

Notepad++

- Esistono vari browser sul mercato (tutti programmi gratuiti):
 - ▶ Internet Explorer: fino a poco tempo fa era il più diffuso,
 - ▶ Mozilla Firefox: ottimo browser ricco di funzionalità e molto aderente agli standard del web
 - ▶ Google Chrome: ottimo browser molto veloce e aderente agli standard del web
 - ▶ Opera: browser norvegese “di nicchia” che ha avuto molto successo sui dispositivi mobili
 - ▶ Safari: browser di Apple molto usato sui Mac

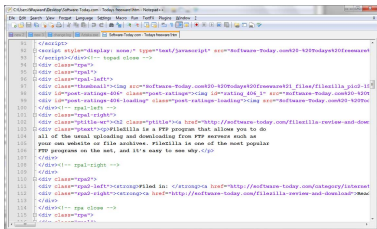
HTML e i browser

- Uno stesso documento HTML **può essere visualizzato in modo diverso da due browser differenti**, a seconda di come i browser interpretano il documento
 - ▶ Esistono degli standard per l'interpretazione, ma non tutti i browser li rispettano (Internet Explorer in particolare)
 - ▶ Spesso è necessario produrre più versioni del documento HTML, una per ogni browser, oppure limitare l'utilizzo di alcuni elementi HTML che notoriamente vengono visualizzati in modo diverso dai principali browser
- Questo discorso sta diventando sempre meno vero...
 - ▶ con la seconda guerra dei browser, l'aderenza agli standard è diventato un fattore di vantaggio competitivo

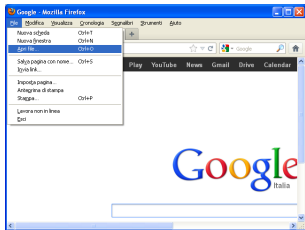
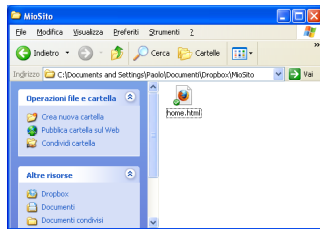
Salvare e visualizzare documenti HTML “in locale” (1)

- Quando si usa un editor (WYSIWYG o testuale) per creare un documento HTML e lo si salva sul proprio computer, bisogna poi usare un browser per visualizzarlo
- Il documento creato **non è su Internet**, ma disponibile solo sul proprio computer (**in locale**)
- Si usa la funzione “Apri file” (o simili) del browser (oppure si fa un doppio click sull'icona del file salvato)

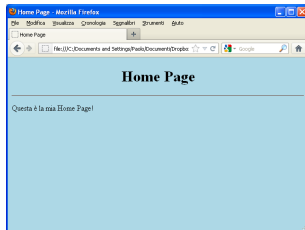
Salvare e visualizzare documenti HTML “in locale” (2)



SALVA

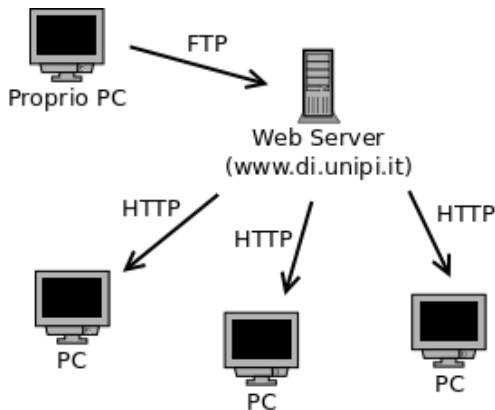


APRI FILE



Pubblicare documenti HTML sul Web

- Per rendere il documento HTML disponibile su Internet (pubblicazione) bisogna copiarlo su un server web
- Di solito si usano programmi basati sul protocollo FTP
- Gli utenti troveranno la pagina sul Web scrivendo il giusto indirizzo sul proprio browser



I tag HTML (1)

- Dal punto di vista umano, un documento di testo può essere strutturato aggiungendo ai contenuti elementi come tabelle, liste, titoli, ecc...
- HTML specifica un insieme di **tag** per strutturare il testo
 - ▶ I browser interpretano i tag HTML e visualizzano la struttura del documento HTML in modo conforme al punto di vista umano
- L'operazione di taggatura (inserimento di tag in un testo) si chiama anche markup

I tag HTML (2)

- Un tag HTML racchiude al suo interno il contenuto da “taggare” ed ha la seguente sintassi

`<tag1>contenuto</tag1>`

- Dove:
 - ▶ `<tag1>`: è l'apertura del tag tag1
 - ▶ `</tag1>`: è la chiusura del tag tag1
 - ▶ `contenuto`: è il contenuto la cui struttura è descritta da tag1
- Ad esempio, il tag `h1` può essere usato così

`<h1>Home Page di Paolo Milazzo</h1>`

per specificare che la frase “Home Page di Paolo Milazzo” è un titolo

- Ad esempio, il tag `p` può essere usato così

`<p>Hello world</p>`

per specificare che la frase “Hello world” è un paragrafo di testo semplice

I tag HTML (3)

- Esistono tag che non prevedono un contenuto
 - ▶ Esempio: Il tag `<hr>` da solo crea una riga orizzontale lungo tutta la pagina
 - ▶ Per questi tag è prevista l'apertura ma non la chiusura (cioè non si usa mai `</hr>`)
- Può capitare che un tag preveda un contenuto ma occasionalmente venga utilizzato senza contenuto
 - ▶ Tag che possono avere un contenuto se usati senza contenuto devono essere scritti in uno dei seguenti modi:
 - ★ `<tag1></tag1>`
 - ★ `<tag1/>`
 - ▶ Anche per `<hr>` si può usare la sintassi equivalente `<hr/>`

I tag HTML (4)

- Un tag HTML può anche prevedere attributi, ovvero delle proprietà specifiche (spesso opzionali) di quel particolare tag
- Gli attributi dei tag possono essere molteplici e ad ognuno si può associare un valore
- Gli attributi si specificano nell'apertura del tag1 con la seguente sintassi

```
<tag1 attr1="valore1" attr2="valore2">contenuto</tag1>
```

- Ad esempio, il contenuto di un paragrafo (tag p) può essere disposto al centro impostando l'attributo align con il valore center, ossia

```
<p align="center">Testo centrato</p>
```

- Le virgolette nel valore associato a un attributo possono essere omesse quanto il valore non contiene spazi bianchi

Annidamento dei tag

- I tag in un documento HTML possono (e spesso devono) essere **annidati** tra loro:
 - ▶ ovvero, un tag può contenere altri tag al suo interno
- Le aperture e chiusure di tag annidati devono essere messi in corrispondenza come se fossero parentesi di un'espressione aritmetica
 - ▶ L'ultimo tag aperto deve essere il primo che si chiude
 - ▶ Esempio: `<tag1><tag2>contenuto</tag2></tag1>` è corretto
 - ▶ Esempio: `<tag1><tag2>contenuto</tag1></tag2>` è SBAGLIATO!!!
- Un testo contenuto dentro a più tag assume tutte le proprietà specificate da tutti i tag
 - ▶ Esempio: `<p>Ciao <u>a tutti</u></p>` specifica che "Ciao a tutti" è un paragrafo di testo in cui "a tutti" deve essere sottolineato (tag u)
- In caso di contraddizioni tra i tag applicati a un contenuto, i tag più interni hanno la priorità

Struttura di un documento HTML (1)

Un documento HTML contiene due parti:

- L'header (intestazione), 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>` all'interno del quale 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>
```

I tag `<html>`, `<head>` e `<body>` annidati come nell'esempio sono **obbligatori** in ogni documento HTML corretto

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>`
- Meta dati (informazioni aggiuntive) sul documento usate dalle applicazioni e dai motori di ricerca. Si inseriscono usando il tag `<meta>`.
- Il tag `<meta>` non prevede un contenuto e ha (di solito) il seguente formato:

```
<meta name="xxx" content="yyy">
```

- ▶ l'attributo `name` specifica il tipo di informazione che si vuole descrivere;
- ▶ l'attributo `content` specifica il contenuto dell'informazione stessa;

L'header di un documento HTML (2)

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 Metodi Informatici per il Turismo">`
- Elencare parole chiave che rappresentano il documento
 - ▶ `<meta name="keywords" content="internet html www http laurea scienze turismo 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) può dare maggiore visibilità alla pagina;
- Attenzione a non 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 (numerate e non numerate)
- Tabelle
- Collegamenti ipertestuali (link ad altri documenti o a altre parti dello stesso documento)
- Immagini
- Mappe (immagini cliccabili)
- Frame (cornici contenenti sotto-documenti)
- Form (moduli compilabili dall'utente)
- Script (piccoli programmi)

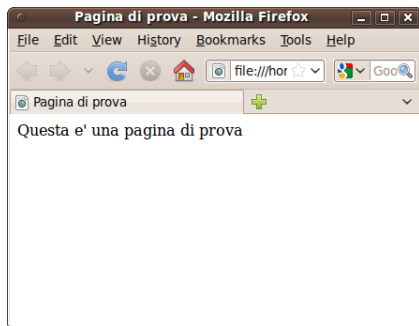
Il corpo di un documento HTML (2)

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>
```



Documenti HTML: note sintattiche (1)

- All'interno di un documento HTML si possono inserire **commenti** (ignorati dal browser) con la seguente sintassi:
`<!-- Questo e' un commento -->`
- I commenti possono occupare più righe del documento
`<!--Questo e' un commento`
che occupa
più righe `-->`

Documenti HTML: note sintattiche (2)

- I commenti servono a chi scrive i documenti HTML per organizzare meglio il codice

```
<html>
  <head>
    <title>Pagina di prova</title>
    <meta name="author"
          content="Paolo Milazzo">
  </head>
  <body>
    <!-- Il corpo della pagina inizia qui -->
    <p>Questa e' una pagina
        di prova</p>
  </body>
</html>
```

Documenti HTML: note sintattiche (3)

- I commenti fanno anche comodo per lasciare nel documento HTML frammenti che non devono (al momento) essere visualizzati

```
<html>
  <head>
    <title>Esercizi</title>
  </head>
  <body>
    <p>Esercizio 1: risolvere 5+4</p>
    <!--
      <p>Soluzione: 9</p>
    -->
  </body>
</html>
```

Documenti HTML: note sintattiche (4)

- I tag html **non sono “case sensitive”**
 - ▶ ovvero, si possono scrivere in maiuscolo o minuscolo indifferentemente: `<html>`, `<HTML>`, `<Html>`,
- HTML collassa tutti i caratteri di whitespace (spazio, tab, a capo, ecc...) in un unico carattere di spazio. Questo permette di organizzare il sorgente in maniera leggibile (indentato) senza influenzare la visualizzazione sul browser
- I seguenti tre esempi sono visualizzati dal browser nello stesso modo

```
<html>
  <head>
    <title>Ciao</title>
  </head>
  <body>
    <p>Prova</p>
  </body>
</html>
```

```
<html><head><title>
Ciao</title></head>
<body><p>Prova</p>
</body></html>
```

```
<html><head>

  <title>
                                Ciao
  </title></head><body>
  <p>
Prova</p>
</body>

</html>
```

Documenti HTML: note sintattiche (5)

- Nei tag HTML è bene utilizzare gli spazi in maniera corretta
- No spazi tra i tag e le parentesi angolari <,>
 - ▶ `<html>` ok, `< html>` no, `<html >` no
- Un singolo spazio tra tag e attributo, e tra coppie di attributi
 - ▶ `<meta name="author" content="Paolo Milazzo">` ok
 - ▶ `<meta     name="author" content="Paolo Milazzo">` no
 - ▶ `<meta name="author"     content="Paolo Milazzo">` no
- Nessuno spazio tra apertura/chiusura di tag e contenuto
 - ▶ `<p>Ciao</p>` ok, `<p>   Ciao</p>` no, `<p>Ciao   </p>` no

Documenti HTML: note sintattiche (6)

- 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
>	>	<	<	&	&	"	"

- Ad esempio, la frase

$1/3 > 1/5$ poiché $3 < 5$

in un documento HTML diventa

$1/3 \> 1/5$ poiché $3 \< 5$

Il corpo di un documento HTML

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

Ci concentreremo sui tag principali (e i relativi attributi). Per maggiori dettagli vedere siti specializzati (ad esempio: www.html.it e www.w3schools.com)

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.

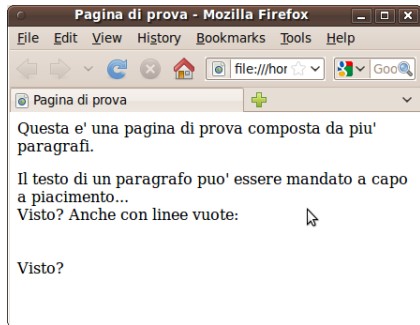
Il testo semplice: paragrafi

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 `<br>` (usato più volte crea righe vuote)

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

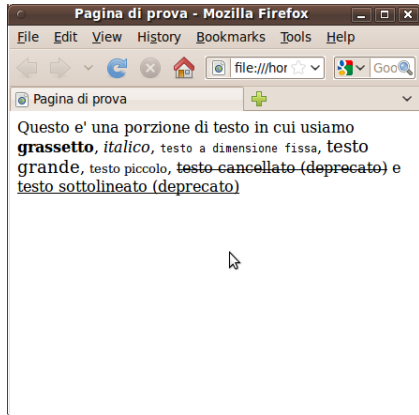
```
<p>Il testo di un paragrafo  
puo' essere mandato a capo  
a piacimento...<br>  
Visto? Anche con linee vuote:  
<br><br><br>  
Visto?  
</p>
```



Il testo semplice: stili (1)

- HTML prevede alcuni stili (grassetto, italico, ecc...) che possono essere assegnati a porzioni di testo, come nei seguenti esempi.
- I tag relativi sono: `<b>`, `<i>`, `<tt>`, `<big>`, `<small>`, `<s>` e `<u>`
- L'uso di alcuni di questi stili è deprecato (si consiglia di usare fogli di stile CSS)

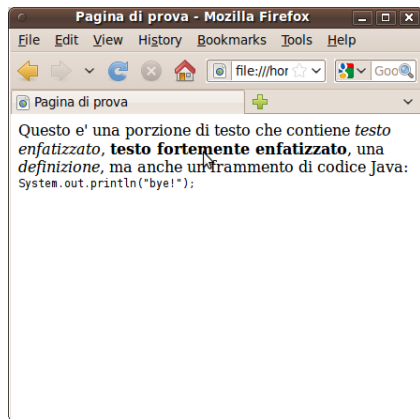
```
<p> Questo e' una porzione di
testo in cui usiamo
<b>grassetto</b>, <i>italico</i>,
<tt>testo a dimensione fissa</tt>,
<big>testo grande</big>,
<small> testo piccolo</small>,
<s>testo cancellato
(deprecato)</s> e <u>testo
sottolineato (deprecato)</u>
</p>
```



Il testo semplice: stili (2)

- HTML prevede anche alcuni stili legati all'interpretazione logica del testo (testo enfatizzato, definizioni,).
- I tag relativi sono: `<em>`, `<strong>`, `<dfn>` e `<code>`
- La visualizzazione (dimensioni del font, ecc...) può dipendere dal browser usato.

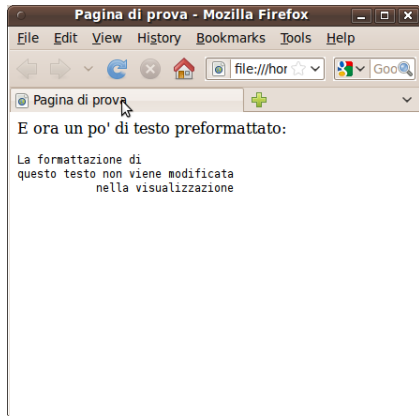
```
<p> Questo e' una porzione di
testo che contiene
<em>testo enfatizzato</em>,
<strong>testo fortemente
enfatizzato</strong>, una
<dfn>definizione</dfn>, ma anche
un frammento di codice Java:
<code>System.out.println("bye!");
</code></p>
```



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 tag relativo è : `<pre>`
- 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.

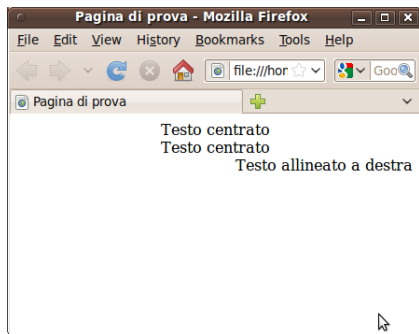
```
<h1>Titolo con h1</h1>  
<h2>Titolo con h2</h2>  
<h3>Titolo con h3</h3>  
<h4>Titolo con h4</h4>  
<h5>Titolo con h5</h5>  
<h6>Titolo con h6</h6>  
<p>Testo semplice</p>
```



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...);
- L'attributo align può essere usato anche con altri tag, quali <p> e <h1>, <h2>, <h3>, ...

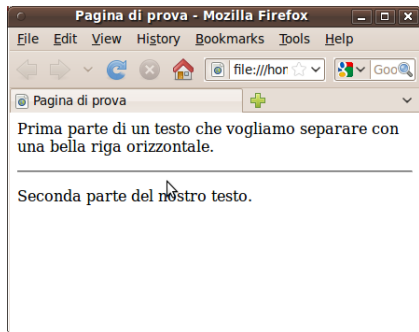
```
<p>  
<center>Testo centrato</center>  
<div align=center>  
Testo centrato</div>  
<div align=right>  
Testo allineato a destra</div>  
</p>
```



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>

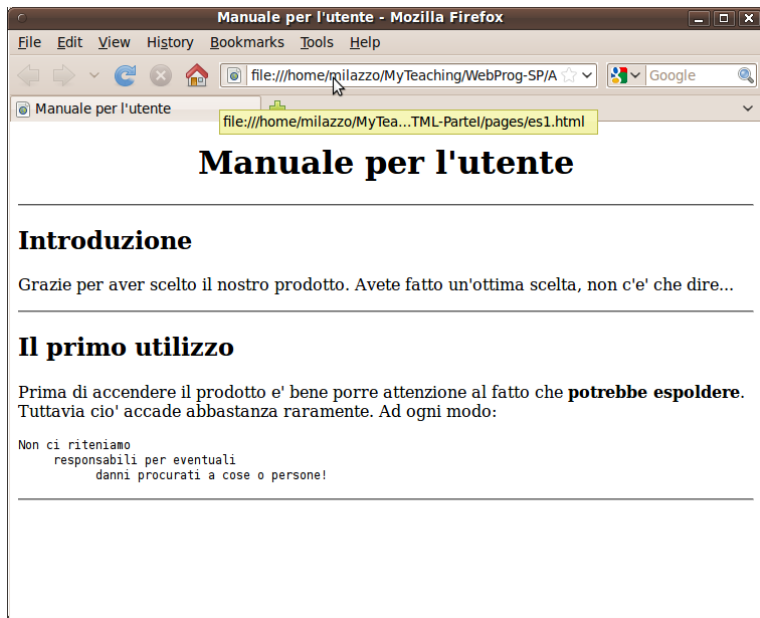
<center><h1>Manuale per l'utente</h1></center>
<hr>

<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)



Le liste (1)

HTML prevede tre tipi di liste:

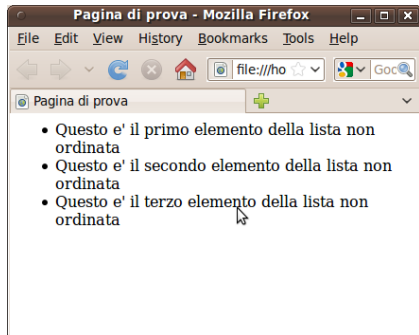
- liste non ordinate (unordered lists)
- liste numerate (ordered lists)
- liste di definizioni (definition lists)

La sintassi delle liste non ordinate e numerate è simile:

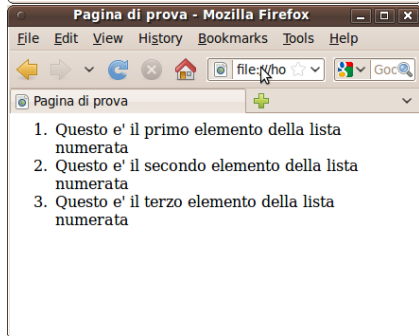
- L'intera lista è racchiusa in `<ul>` (per liste non ordinate) o in `<ol>` (per liste numerate);
- Ogni elemento è racchiuso in `<li>` (per entrambi i tipi di lista) – `</li>` può essere omesso.

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>
```

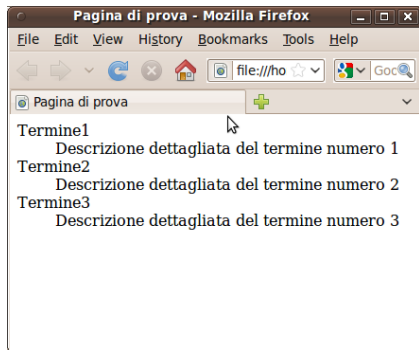


Le liste (3)

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>
```



Riferimenti URL

- Il contenuto di un documento HTML contiene spesso riferimenti a risorse esterne
 - ▶ immagini (o altri contenuti multimediali) da visualizzare nella stessa pagina web
 - ▶ link ad altri documenti HTML (altre pagine dello stesso sito o pagine di siti diversi)
- Per fare riferimento a una risorsa esterna si usa un URL (o indirizzo)
- Due tipi di URL:
 - ▶ **URL assoluto** per riferire a risorse (immagini o pagine) di siti diversi
 - ▶ **URL relativo** per riferire a risorse (immagini o pagine) dello stesso sito

Riferimenti URL

- Il contenuto di un documento HTML contiene spesso riferimenti a risorse esterne
 - ▶ immagini (o altri contenuti multimediali) da visualizzare nella stessa pagina web
 - ▶ link ad altri documenti HTML (altre pagine dello stesso sito o pagine di siti diversi)
- Per fare riferimento a una risorsa esterna (e.g. un file) si usa un URL (o indirizzo)
- Due tipi di URL:
 - ▶ **URL assoluto** per riferire a file (immagini o pagine) di siti diversi
 - ▶ **URL relativo** per riferire a file (immagini o pagine) dello stesso sito

URL assoluti

- Un URL assoluto è un URL completo di tutte le sue parti
- Esempio:

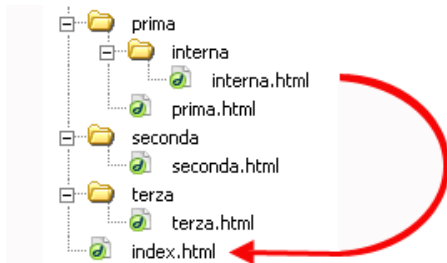
`http://www.di.unipi.it/corsi/AA1112/index.html`

- consiste di:
 - ▶ protocollo usato per richiedere il file (`http://`)
 - ▶ sito (e quindi server) dove trovare il file (`www.di.unipi.it`)
 - ▶ la directory (cartella) del sito che contiene il file (`/corsi/AA1112/`)
 - ▶ il nome del file (`index.html`)

URL relativi

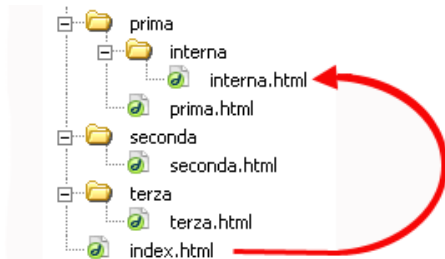
- Un URL relativo si usa solo per fare riferimento a file dello stesso sito che contiene il documento HTML corrente
- Un URL relativo descrive in percorso da fare per raggiungere un file **a partire dalla directory che contiene il documento corrente**
- Se il file da raggiungere è nella stessa directory del documento HTML corrente, l'URL relativo coincide con il nome del file
 - ▶ Esempio: `index.html`

URL relativi: esempi (1)



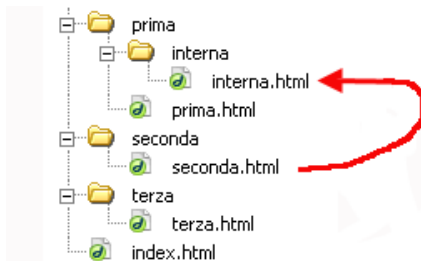
- Se nel documento `interna.html` vogliamo fare riferimento al file `index.html` è necessario considerare che tale file si trova nella directory principale del sito che è due livelli sopra la directory di `interna.html`
- Per spostarsi alla directory superiore si usa il percorso `../`
- L'URL relativo in questo caso è `../../index.html`

URL relativi: esempi (2)



- Viceversa, se nel documento `index.html` vogliamo fare riferimento al file `interna.html` è necessario considerare che tale file si trova nella directory principale del sito che è due livelli sotto la directory di `index.html`
- Per spostarsi alle directory inferiori si scrivono i loro nomi nell'ordine e separati da /
- L'URL relativo in questo caso è `prima/interna/interna.html`

URL relativi: esempi (3)



- Se invece nel documento `seconda.html` vogliamo fare riferimento al file `interna.html` è necessario considerare che raggiungere tale file bisogna prima salire di un livello e poi scendere di due
- L'URL relativo in questo caso è `../prima/interna/interna.html`

Le immagini (1)

- Un'immagine può essere inserita tramite il tag `<img>`
 - ▶ I formati di immagine accettati sono solitamente GIF e JPEG
- Il tag `<img>` prevede solo attributi (non un contenuto) ed ha il formato ``
 - ▶ 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.

```

```



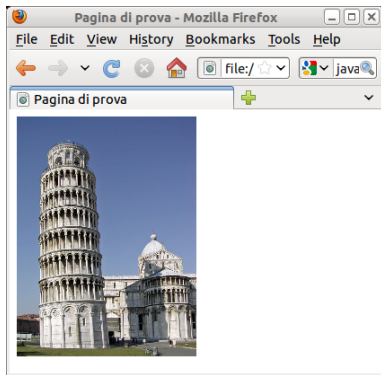
Le immagini (2)

- E' anche possibile specificare le dimensioni di un'immagine (larghezza e altezza) usando gli attributi `width` e `height`
- Specificando solo uno dei due attributi l'altro viene impostato automaticamente in modo da preservare le proporzioni dell'immagine (rapporto altezza/larghezza)

- Tali attributi prendono (di default) valori espressi in pixels come segue:
``.

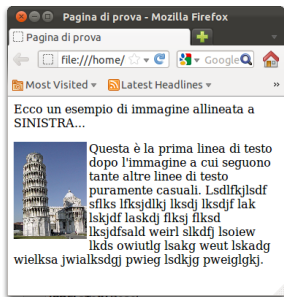
```

```



Le immagini (3)

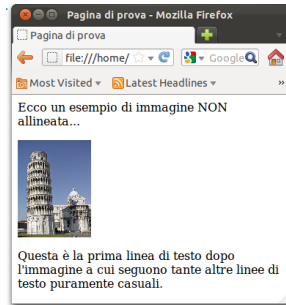
- E' possibile specificare la posizione dell'immagine rispetto al testo della pagina web
- Per fare questo si utilizza l'attributo `align` che può assumere i seguenti valori:
 - ▶ `left`, `right`: allinea l'immagine a sinistra o a destra del testo
- Se non si utilizza l'attributo `align` l'immagine interrompe il testo



`align="left"`



`align="right"`



`align non usato`

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 i 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

Riepilogo dei tag trattati

- Tag trattati in questa lezione:

- ▶ `<html>`, `<head>`, `<body>`
- ▶ `<title>`, `<meta>`
- ▶ `<p>`, `<br>`
- ▶ `<b>`, `<i>`, `<tt>`, `<big>`, `<small>`, `<s>`, `<u>`
- ▶ `<em>`, `<strong>`, `<dfn>`, `<code>`
- ▶ `<pre>`
- ▶ `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, `<h6>`
- ▶ `<center>`, `<div>`
- ▶ `<hr>`
- ▶ `<ul>`, `<ol>`, `<li>`, `<dl>`, `<dt>`, `<dd>`
- ▶ `<img>`
- ▶ `<a>`