

(X)HTML

[Oreste Signore](mailto:oreste@w3.org), <oreste@w3.org>

Responsabile Ufficio Italiano W3C

Area della Ricerca CNR - via Moruzzi, 1 - 56124 Pisa

[Master in Comunicazione e New Media](#)

[Ateneo Pontificio Regina Apostolorum](#)

Corso: [Architettura del Web](#)

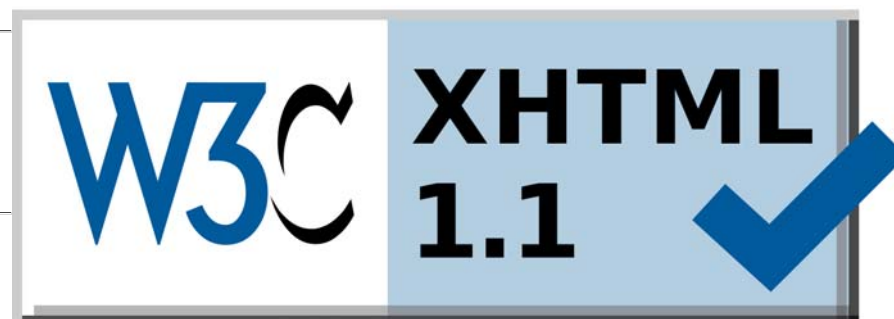
Presentazione: <http://www.w3c.it/education/2012/upra/html/>

Versione pdf: <http://www.w3c.it/education/2012/upra/html.pdf>

Formato XHTML realizzato usando il tool [Slidy](#) di Dave Raggett.

[Slidy](#) dovrebbe funzionare in tutti i browser moderni con Javascript abilitato. Usare freccia destra/sinistra per muoversi da una slide all' altra.

Vedi [la pagina di aiuto di Slidy](#) per ulteriori informazioni.



WORLD WIDE WEB
consortium®
Ufficio Italiano



Contenuto

- ***Introduzione***

- [Struttura di base](#)

- [Liste](#)

- [Link](#)

- [Altri elementi block-level](#)

- [Altri elementi inline](#)

- [Immagini](#)

- [Tabelle](#)

- [Moduli \(forms\)](#)

- [Head del documento](#)

Domande!!!

alzi la mano chi sa:

- cosa significa:

```
<ol>
  <li>primo
  <li>secondo
</ol>
```

- se il codice scritto sopra è errato
- spiegarmi perché
- 😊
- prime domande per sciogliere il ghiaccio, iniziamo
- questa è la versione compressa di un corso compresso...

Tecnologie assistive

- Possono aiutare molte persone, non solo quelle considerate disabili (o, meglio, *persone con funzionalità ridotte*)
TUTTI gli utenti hanno dei *benefici* se i documenti sono *accessibili*
- nuovi studi di ricerca mostrano che il 57 per cento degli utenti adulti di computer possono avere benefici dalla tecnologia accessibile

(Forrester Data)



Le varie versioni di HTML

- HTML: Hyper Text Markup Language
 - *HTML 2.0*
 - *HTML 3.2 (1996, prima W3C Rec)*
 - *HTML 4.0*
 - *HTML 4.01*
- XHTML: eXtensible Hyper Text Markup Language
 - *XHTML 1.0 (riformulazione di HTML in XML, 2000, update nel 2002)*
 - *XHTML 1.1 (modularizzazione di XHTML 1.0, individuazione dei costrutti "deprecated"), 2001*
 - *XHTML-Print, (stampa da dispositivi mobili e su stampanti a basso costo), 2006*
 - *XHTML 1.1 Basic (versione minima per dispositivi mobili e PDA), 2008*
 - *XHTML 2.0 (Working Draft) al contrario di XHTML 1.x non pensato per essere compatibile con le versioni precedenti (HTML 4.01, XHTML 1.x) dichiarato **chiuso**, confluito (in parte) in HTML5*
- HTML5

Contenuto

- [Introduzione](#)
- **[Struttura di base](#)**
- [Liste](#)
- [Link](#)
- [Altri elementi block-level](#)
- [Altri elementi inline](#)
- [Immagini](#)
- [Tabelle](#)
- [Moduli \(forms\)](#)
- [Head del documento](#)

Elementi e attributi

- L' informazione viene completata con un *markup*
- Il markup descrive la *struttura*
- L' informazione marcata è costituita da *elementi*

elemento

```
<tagname>contenuto dell' elemento</tagname>
```

start tag: `<tagname>`

end tag: `</tagname>`

può contenere altri elementi:

```
<elemento>questo elemento contiene un altro  
<elementoInterno>elemento</elementoInterno>  
che deve essere codificato in modo corretto</elemento>
```

può avere degli *attributi*

attributi

codificati nello *start tag* (attributeName = attributeValue)

valore dell' attributo racchiuso tra apici (semplici o doppi)

```
<elemento attr1="valAtt1" attr2='valAtt2'>elemento con attributi</elemento>
```

Come è fatto un documento HTML

- Dichiarazione del *tipo di documento* (XML, DTD)

- *head*

- *metainformazioni sul documento*
- *titolo*
- *stili*

- *body*

- *contenuto del documento*

- memorizzato con estensione **.html** (o **.htm**)

- *Dichiarazione DOCTYPE non è un elemento XHTML, e non dovrebbe avere un tag di chiusura*

- un [esempio](#)

```
<!DOCTYPE ....Dichiarazione del tipo di documento (DTD).... >
<html>
  <head>
    <title> titolo del documento </title>
    ... metainformazioni sul documento ...
    ... stili ...
  </head>
  <body>
    ...contenuto del documento...
  </body>
</html>
```

Nota per l'accessibilità

importante fornire un titolo (*title*) descrittivo per tutte le pagine Web

può essere utile anche per indicare in che posizione del sito ci troviamo:

```
<title>Home :: Descrizione Livello 1 :: Livello 2 :: Descrizione pagina </title>
```

Differenze principali tra HTML e XHTML

- I documenti XHTML devono essere ben formati
- Gli elementi del documento XHTML devono essere annidati correttamente
- Minuscole e maiuscole sono interpretate in modo differente
- Tutti gli elementi di XHTML devono essere chiusi

I documenti devono essere ben-formati

rivedremo meglio con XML

- Tutti gli elementi di XHTML devono essere annidati all'interno dell'elemento radice `<html>`
- Tutti gli elementi devono essere chiusi
- Tutti gli altri elementi possono avere elementi sotto-elementi (figli)
- I sotto-elementi devono essere annidati correttamente all'interno del loro elemento genitore
- Tutti i documenti di XHTML devono avere una dichiarazione di DOCTYPE. Gli elementi html, head, body devono essere presenti, title deve essere presente all'interno dell'elemento head

Corretto annidamento degli elementi

Errato:

```
<strong><em>testo </strong></em>
```

Corretto:

```
<strong><em>testo </em></strong>
```

Interpretazione minuscole e maiuscole

- XML è case-sensitive (le lettere minuscole vengono interpretate diversamente dalle maiuscole)
- I tag come <p> e <P> sono interpretati in modo diverso

Errato:

```
<BODY>  
  <P>Questo è un paragrafo</P>  
</BODY>
```

Corretto:

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

Tutti gli elementi XHTML devono essere chiusi

Gli elementi devono avere un tag di chiusura

Errato:

```
<p>Questo è un paragrafo  
<p>Questo è un altro paragrafo
```

Corretto:

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

Anche gli elementi vuoti devono essere chiusi

Anche gli elementi vuoti devono avere un tag di chiusura oppure il tag di apertura deve finire con "/>"

Errato:

```
Questa è una interruzione <br>
Questa è una linea orizzontale <hr>
Questa è una immagine 
```

Corretto:

```
Questa è una interruzione <br />
Questa è una linea orizzontale <hr />
Questa è una immagine 
```

Per compatibilità con tutti i browser, aggiungere uno spazio supplementare prima del simbolo "/"
come
 e <hr />

Elementi inline e block-level

inline

- possono contenere solo testo o altri elementi di tipo inline
- `` e `` sono esempi di elementi inline

block-level

- elementi più importanti, possono definire lo spazio occupato, e possono contenere altri elementi di tipo inline
- `<p>` e `<h1>` sono esempi di elementi block-level
- alcuni elementi block-level possono contenere altri elementi block-level
- in genere gli elementi block-level iniziano su una nuova linea

Esercizio

Con un editore testi (notepad, gedit, vi, ...) scrivere un documento XHTML 1.1

Il corpo del documento deve essere misto, oltre a del testo deve contenere i seguenti elementi:

- p (paragrafo)
- em (enfasi)
- strong (enfasi forte)
- br (break line, a capo)
- hr (horizontal row, linea orizzontale)

Ricordarsi di dichiarare il DTD

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

Commenti

Possono apparire dovunque in un documento HTML

Esempio:

```
<!-- Questo è un commento -->
```

Header e paragrafi

Header

- sei livelli di intestazione (da **h1** a **h6**)
- di importanza decrescente
- normalmente visualizzati con dimensioni del carattere decrescenti e altri accorgimenti tipografici
- è opportuno utilizzarli in sequenza
- devono corrispondere a una struttura logica del documento
- non sono un mero accorgimento tipografico
- un [esempio](#)

Paragrafo

- `<p>Contenuto del paragrafo</p>`

Alcuni attributi comuni a quasi tutti gli elementi

Identificazione e stile

- **id=** identificatore univoco
- **class=** una o più classi
- **style=** in genere lo stile viene definito a livello globale, ma talvolta può essere utile definirlo per una singola istanza (*non è in generale una buona prassi*)
- **title=** informazione aggiuntiva sull' elemento. Normalmente i browser utilizzano questa informazione (è anche un elemento importante per l' accessibilità)

Lingua

- **lang=**
- **xml:lang=**

Interazione

- **onfocus** (onactivate) (evitare interazioni dipendenti da dispositivi come onmouseover, onmouseout, and ondblclick)
- **onblur** (opposto di onfocus)
- **onclick** caso particolare: se non usato con link controlli di form, è dipendente dal dispositivo ed è pertanto evitare o associare anche altre interazioni dipendenti da dispositivi (onKeyPress)
- **onmousedown=**

L'attributo xml:lang

- Si applica a quasi tutti gli elementi XHTML
- Specifica il linguaggio del contenuto di un elemento

```
<div xml:lang="en">Hello!</div>
```

Accessibilità: specificare ogni cambiamento di lingua (...se si tratta di una sola parola di uso comune evitare, considerare che gli screen reader ci mettono del tempo a caricare i fonemi di un'altra lingua)

Riferimenti a Entità

- Possibile utilizzare un riferimento ad un valore esadecimale, per esempio per i caratteri diacritici o per specificare i colori
- Quando esistono, si possono utilizzare le parole chiave
- Altro metodo consente di utilizzare il codice esadecimale, usando `&xnumero;` (x e nn in minuscolo)
Esempio `&xfffff;` (bianco)
- HTML definisce un certo numero di entità per quei caratteri che sono:
 - *proibiti perché usati in HTML (<, >, &, ", ecc.)*
 - *non presenti nell'ASCII a 7 bit*

Il browser sa come rappresentarle

- `&`; &
- `<`; (less than) <
- `®`; ®
- `À` À
- `æ` æ
- `à` à
- `ç` ç
- `ñ` ñ
- `"`; "
- `>`; (greater than) >
- ` `; (non-breaking space)
- `Á` Á
- `Ä` Ä
- `á` á
- `ä` ä
- ...

Enfasi

em

- normalmente visualizzato in *corsivo*

strong

- normalmente visualizzato in **grassetto**

L' unica assunzione ragionevole è che il browser visualizzi con diversi livelli di importanza

Il break

- Gli elementi block-level ignorano gli spazi addizionali e la formattazione originale del testo
- Se in un paragrafo si vuole forzare il salto a nuova riga, si può usare l' elemento inline `
`
- Si noti la codifica (*empty element*)

Contenuto

- [Introduzione](#)
- [Struttura di base](#)
- ***Liste***
- [Link](#)
- [Altri elementi block-level](#)
- [Altri elementi inline](#)
- [Immagini](#)
- [Tabelle](#)
- [Moduli \(forms\)](#)
- [Head del documento](#)

Liste ordinate e non ordinate

Liste non numerate (*unordered list*)

- sequenza di punti aventi la stessa importanza
- `` inizio lista
- `` fine lista
- all' interno di un elemento `` *sono ammessi solo* elementi ``

[Un esempio](#)

Liste numerate (*ordered list*)

- sequenza di punti di cui si ritiene importante l' ordine, o in cui comunque risulti utile identificare i singoli elementi
- `` inizio lista
- `` fine lista
- all' interno di un elemento `` *sono ammessi solo* elementi ``

L' elemento di lista (*list item*)

- ``punto della lista``
- può contenere *qualsunque elemento block-level* (quindi anche `<h1>` o `<p>`)
- per inserire una lista in un' altra lista:

```
<li>
  <ul>
    <li></li>
  </ul>
</li>
```

Liste di definizione

I tag

- `<dl>` e `</dl>`: racchiudono la “definition list”
- `<dt>` e `</dt>`: identificano il termine da definire
- `<dd>` e `</dd>`: racchiudono la definizione del termine.

Le regole

- Si possono avere elementi `<dt>` senza elementi `<dd>` e viceversa, ma non elementi privi di contenuto
- Si possono avere più elementi `<dd>` per un elemento `<dt>`, se un termine ha più definizioni
- Un elemento `<dd>` può contenere qualunque combinazione di elementi inline e block-level

Un esempio

```
<dl>
  <dt>v&egrave;nto</dt>
  <dd>movimento di masse d'aria</dd>
  <dt>tappo</dt>
  <dd>accessorio impiegato per la chiusura di contenitori</dd>
  <dd>persona di bassa/breve statura</dd>
</dl>
```

vènto

movimento di masse d'aria

tappo

accessorio impiegato per la chiusura di contenitori

persona di bassa/breve statura

Contenuto

- [Introduzione](#)
- [Struttura di base](#)
- [Liste](#)
- **[Link](#)**
- [Altri elementi block-level](#)
- [Altri elementi inline](#)
- [Immagini](#)
- [Tabelle](#)
- [Moduli \(forms\)](#)
- [Head del documento](#)

Link

Cosa sono

- Sono l'essenza del web e la ragione del suo successo
- Un link permette di collegare un documento ad un altro, o anche ad una sua parte specifica, o anche ad un altro servizio
- I link possono rinviare ad un altro documento sullo stesso server o su qualunque altro server nel mondo

Come appaiono

- Normalmente rappresentati dal browser con un [testo sottolineato](#)
- Il loro aspetto può essere determinato tramite CSS

I tag

- ``: definisce la destinazione del link
- attributo `id`: identifica il punto del documento a cui salta il link, e che verrà reso visibile (opzione molto utile quando si salta ad un documento lungo). Es.: `<h1 id="capitolo3">`
- `<a hreflang="<en|it|...>">`: identifica la lingua della risorsa destinazione del link. Importante per l'accessibilità

Uniform Resource Identifier (URI)

- Il valore dell' attributo `href` deve essere un *Uniform Resource Locator* o, meglio, un *Uniform Resource Identifier* (URI)
- Definisce un *indirizzo* Internet, il *protocollo* da utilizzare e la *posizione* del documento
- È costituito da:
 - *protocollo* (*http, ftp, news...*)
 - *host* (*macchina.dominio.paese*)
 - *port* (*normalmente non necessario*)
 - *path*
 - *filename*
- Un URI definisce l'*identità* di risorse sul Web

Esempi:

`protocollo://server:porta/percorso/nomedelfile`

`http://orario.isti.cnr.it:18080/Orario/jsp/index.jsp`

URI, URL, URN

- *URL* (Uniform Resource Locator) e *URN* (Uniform Resource Name) sono *URI*
- *URN*: definisce un nome, non fornisce dettagli su come ottenere la risorsa su una rete
- *URL*: definisce come recuperare la risorsa sul Web

Cool URI don't change!

URL relativi

- Non contengono il protocollo o le informazioni sull' host
- In genere fanno riferimento a documenti residenti sullo stesso host su cui risiede il documento origine del link
- `xyz.html` definisce un file nella stessa directory della pagina corrente
- `..` indica la directory padre della directory corrente
- Supponendo che la pagina corrente sia nella directory *path*
`../path2/filename`
path2 e *path* sono directory allo stesso livello

Esempi di link

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="it" lang="it">
  <head>
    <title>HTML-Linking</title>
    [...]
  </head>
  <body>
    <h1 id="indice">Indice</h1>
    <ol >
      <li><a href="#id1">Primo paragrafo</a></li>
      <li><a href="#id2">Secondo paragrafo</a></li>
      [...]
    </ol>
    <h1 id="id1">Primo paragrafo</h1>
    <p>Questo è un paragrafo che [...]</p>
    <p>È utile poter avere sempre la possibilità di
      tornare all' <a href="#indice">indice</a></p>
    <h1 id="id2">Secondo paragrafo</h1>
    [...]
  </body>
</html>
```

[Vai all' esempio completo](#)

Accessibilità dei link (1)

title

l'attributo **title** è utile per chiarire lo scopo del collegamento

```
<a href="./riferimento.html">clicca qui</a>  
// doppiamente errato: dipendente da dispositivo, non descrittivo
```

aiuto

Utile un collegamento ad una pagina di aiuto in ogni pagina, (in cui elencare le accesskey...)

```
<a href="aiuto.html">Help</a>
```



Accessibilità dei link (2)

per disabili visivi

Utile un link ad inizio pagina per saltare direttamente al contenuto "reale"
(esigenze di stile? con il CSS il collegamento può essere spostato in una zona non visibile...)

```
<div id="salta_al_contenuto"><a href="#contenuto">Salta al contenuto principale</a></div>
... intestazione
... navigazione
<div id="contenuto">
</div>
```

per gli screen reader

lingua della pagina riferita scritta differente da quella della pagina corrente

```
<a hreflang="en" href="...">....</a>
```

per disabili motori

separare link adiacenti (strutturalmente, ad es. con liste) per facilitare la selezione del link voluto

```
<h2>Argomenti</h2>
<ul class="navigazione">
  <li><a href="romanzi.html">Romanzi</a></li>
  <li><a href="fantascienza.html">Fantascienza</a></li>
  <li><a href="gialli.html">Gialli</a></li>
</ul>
```

Contenuto

- [Introduzione](#)
- [Struttura di base](#)
- [Liste](#)
- [Link](#)
- **[*Altri elementi block-level*](#)**
- [Altri elementi inline](#)
- [Immagini](#)
- [Tabelle](#)
- [Moduli \(forms\)](#)
- [Head del documento](#)

Testo preformattato

Tag

- `<pre>`

Effetto

- Il testo viene formattato esattamente come nella codifica originale, preservando spazi, ritorni a capo, etc.
- Usato spesso per includere nelle pagine web codice sorgente

Address

Tag

- `<address>`

Effetto

- Eredità dei primi browser
- Il suo effetto potrebbe essere ottenuto con un `<p class="address">`
- Normalmente reso dal browser con testo in corsivo: es.

```
<address>
  Oreste Signore<br />
  CNR-ISTI
</address>
```

Viene reso come:

Oreste Signore
CNR-ISTI

Citazioni

Motivazioni

Evidenziare testo proveniente da altre fonti (importanza del copyright nel web, dove è facile integrare contenuti prodotti da altri)

Tag

- `<blockquote cite=" ">`

Esempio

```
<p>Il ruolo del W3C è:</p>
<blockquote cite="http://www.w3.org" >
  <p>The World Wide Web Consortium ( <acronym title="World Wide Web Consortium">W3C</acronym> )
  develops interoperable technologies (specifications, guidelines, software,
  and tools) to lead the Web to its full potential.
  W3C is a forum for information, commerce, communication, and collective
  understanding. On this page, you'll find [...]</p>
</blockquote>
```

Effetto

Il ruolo del W3C è:

The World Wide Web Consortium (W3C) develops interoperable technologies (specifications, guidelines, software, and tools) to lead the Web to its full potential. W3C is a forum for information, commerce, communication, and collective understanding. On this page, you'll find [...]

Nota

Non esiste un meccanismo per vedere il valore dell' attributo `cite`, se non l' esame diretto del codice sorgente della pagina

Riga orizzontale

Motivazioni

Evidenziare e separare contenuti nella pagina

Tag

- `<hr />`

Esempio

```
<p>Prima notizia</p>
<hr />
<p>Seconda notizia</p>
```

Effetto

Prima notizia

Seconda notizia

Nota

È più un elemento stilistico che di contenuto

L' elemento generico <div>

Motivazioni

Serve a raggruppare un insieme di elementi

Tag

`<div>`

Note

- È un generico elemento block-level
- Può contenere elementi block-level e inline
- Può avere un attributo `class`
- Può avere un attributo `id`
- Molto potente per l' impaginazione e lo stile della pagina (ampiamente usato nei CSS)

Contenuto

- [Introduzione](#)
- [Struttura di base](#)
- [Liste](#)
- [Link](#)
- [Altri elementi block-level](#)
- **[Altri elementi inline](#)**
- [Immagini](#)
- [Tabelle](#)
- [Moduli \(forms\)](#)
- [Head del documento](#)

Elementi importanti della frase

- Sono elementi inline che definiscono la semantica del testo racchiuso tra i tag.
- Possono essere utilizzati dal browser (che li presenterà in maniera diversa), ma anche da spell-checker o motori di ricerca e indicizzazione
- **cite** citazione o riferimento ad altro documento
- **dfn** punto di definizione del termine
- **code** frammento di codice
- **samp** esempio di output di programmi, script, etc.
- **var** argomento o variabile di un programma
- **abbr** abbreviazione (usa l' attributo title per definire l' abbreviazione)
- **acronym** acronimo (usa l' attributo title per definire l' acronimo)

Matematica

- Non è particolarmente sviluppato
- Esiste MathML, un apposito linguaggio di markup
- È possibile codificare *apici* (elemento <sup>) e *pedici* (elemento <sub>)

```
a<sub>i, j</sub>=k*x<sup>i</sup>*y<sup>j</sup>
```

```
 $a_{i,j}=k*x^i*y^j$ 
```

Citazioni

Motivazioni

Ha la stessa funzione dell' elemento **blockquote**, ma consente di codificare citazioni come elementi *inline* (nel paragrafo corrente)

Tag

```
<q cite="">
```

Esempio

```
<p>Oreste disse <q cite="http://www.weblab.isti.cnr.it/people/oreste">
Mi sembra che Dante abbia detto
<q cite="http://it.wikipedia.org/wiki/Dante_Alighieri">Non ti curar di loro, ma guarda e passa</q>
e che questa sia un' ottima massima</q>.</p>
```

Effetto

Oreste disse *“Mi sembra che Dante abbia detto ‘Non ti curar di loro, ma guarda e passa’ e che questa sia un' ottima massima”*.

Note

- Il browser dovrebbe introdurre livelli diversi di apici
- Poco supportato:
 - *IE non introduce virgolette*
 - *Opera usa lo stesso tipo di virgolette*

L' elemento generico

Motivazioni

Serve a raggruppare del testo ed elementi inline in modo da attribuirgli un comportamento globale

Tag

Note

- È un generico elemento *inline*
- Può contenere qualunque elemento inline
- Può avere un attributo **class**
- Può avere un attributo **id**
- Molto potente per l' impaginazione e lo stile della pagina (ampiamente usato nei CSS)

Contenuto

- [Introduzione](#)
- [Struttura di base](#)
- [Liste](#)
- [Link](#)
- [Altri elementi block-level](#)
- [Altri elementi inline](#)
- **[Immagini](#)**
- [Tabelle](#)
- [Moduli \(forms\)](#)
- [Head del documento](#)

 e <object>

- È il meccanismo più semplice
- Non fornisce molte alternative se l' immagine non è disponibile
- Era previsto l' abbandono di questo tag

<object>

- Consente di utilizzare alternative
- Molto utile quando sono richieste particolari funzionalità al browser

Immagini inline

- Il tag `` è un “*empty element*”
- Le immagini possono essere posizionate con uno *style sheet*

Gli attributi

src

- Specifica un URL per l' immagine da mostrare
- l' URL può essere assoluto o relativo
- Attributo obbligatorio per ogni elemento ``

alt

- Descrizione testuale dell' immagine
- Testo che compare se l' immagine non è disponibile
- I browser puramente testuali (o i browser vocali) restituiscono il contenuto di questo attributo
- È un attributo obbligatorio

title

- quando il mouse passa sull'immagine, viene restituito il contenuto di questo attributo
- i vari browser si comportano in maniera diversa

Specifica delle dimensioni dell' immagine

height

- Specifica l' *altezza* desiderata dell' immagine in pixel (può essere indicata anche in percentuale)
- Meglio specificarlo, quando è noto

width

- Specifica la *larghezza* desiderata dell' immagine in pixel (può essere indicata anche in percentuale)
- Meglio specificarlo, quando è noto

max-width (o max-height)

- Specifica la *larghezza* (o l' *altezza*) *massima* dell' immagine in pixel (può essere indicata anche in percentuale)
- Meglio specificarlo, per evitare la sgranatura dell' immagine o cattive impaginazioni.


Nota

- Se vengono fornite altezza e larghezza, il browser riserva lo spazio necessario per l' immagine prima che venga scaricata dalla rete
- Rende più rapida la presentazione dei contenuti

Un esempio: inline con dimensioni esatte

```
<p>  
  
    The following image is of a happy face image defined in  
    the gif image format:  
      
    . The image element is a character-level or inline element  
    which means  
    that it appears like a rather large letter or word  
    in the middle of the paragraph. Character-level  
    elements  
    form part of the text flow within the paragraph.  
    Control of the image's position is usually  
    done with a style sheet.  
</p>
```




The following image is of a happy face image defined in the gif image format: . The image element is a character-level or inline element which means that it appears like a rather large letter or word in the middle of the paragraph. Character-level elements form part of the text flow within the paragraph. Control of the image's position is usually done with a style sheet.

Un esempio: inline con dimensioni ingrandite

```
<p>  
  
    The following image is of a happy face image defined in  
    the gif image format:  
      
    The image element is a character-level or inline element  
    which means  
    that it appears like a rather large letter or word  
    in the middle of the paragraph. Character-level  
    elements  
    form part of the text flow within the paragraph.  
    Control of the image's position is usually  
    done with a style sheet.  
</p>
```



The following image is of a happy face image defined in the gif image format: . The image element is a character-level or inline element which means that it appears like a rather large letter or word in the middle of the paragraph. Character-level elements form part of the text flow within the paragraph. Control of the image's position is usually done with a style sheet.

Object

Motivazioni

- Permette di specificare varie alternative

Tag e (alcuni) attributi

- **object** definisce uno di un insieme di elementi
- **data=** specifica l' URL
- **type=** specifica il MIME-type
- **standby=** specifica un messaggio da presentare mentre viene scaricata l' immagine
- **height=**, **width=**, **max-width=**, **alt=**, etc. come al solito

Esempio

```
<object width="349" height="91" data="bobbin.jpg" type="image/jpeg"
standby="JPEG image downloading">
<object width="349" height="91" data="bobbin.png" type="image/png"
standby="PNG image downloading" />

No image available
</object>
```

Effetto

- Se il primo tipo di object non è disponibile, prova con il secondo, e così via

Contenuto

- [Introduzione](#)
- [Struttura di base](#)
- [Liste](#)
- [Link](#)
- [Altri elementi block-level](#)
- [Altri elementi inline](#)
- [Immagini](#)
- **[Tabelle](#)**
- [Moduli \(forms\)](#)
- [Head del documento](#)

Tabelle

Questo argomento non viene trattato in dettaglio

- *Utilizzare per rappresentare tabelle di dati, NON per impaginare*
- Le celle possono contenere testo, immagini, collegamenti o *altre tabelle*

Formato di base:

```
<table summary="Questa è una tabella a due colonne, con una intestazione. Leggere le righe da sinistra a destra">
  <caption>Titolo tabella</caption>
  <thead>
    <tr>
      <th>Prima intestazione</th><th>Seconda intestazione</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Prima riga prima colonna</td><td>prima riga seconda colonna</td>
    </tr>
    <tr>
      <td>Seconda riga prima colonna</td><td>Seconda riga seconda colonna</td>
    </tr>
  </tbody>
</table>
```

attributo *summary*

Fornire un sommario relativo allo scopo e alla struttura per i browser che rappresentano media non visuali

elemento *caption*

Fornire un titolo

Contenuto

- [Introduzione](#)
- [Struttura di base](#)
- [Liste](#)
- [Link](#)
- [Altri elementi block-level](#)
- [Altri elementi inline](#)
- [Immagini](#)
- [Tabelle](#)
- **[Moduli \(forms\)](#)**
- [Head del documento](#)

Moduli (forms)

- Utilizzati per far inserire informazioni e valori da spedire al server per elaborazioni
- Il browser:
 - *raccolge i dati inseriti dall'utente*
 - *crea una connessione HTTP e invia le informazioni al destinatario specificato nell'attributo ACTION del form*
- Il destinatario:
 - *riceve i dati*
 - *li elabora e genera un documento di risposta, che viene spedito, tramite il server HTTP, al browser*
- Un form può prevedere: campi di inserimento dati, pulsanti, bottoni radio, checkbox, liste a scomparsa, ecc.

Un esempio di form

- Il [form](#)...
- ...e il codice

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Form XHTML</title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <script type="text/javascript" src="validaform.js"></script>
  </head>
  <body>
    <form action="http://www.regione.toscana.it:8080/servlet/Controllore" method="post"
      enctype="multipart/form-data" >
      <p><input type="hidden" name="nascosto" value="v" />
      <label for="nome">Text</label> <input type="text" id="nome" name="nome" /></p>
      <p><label>Password <input type="password" name="password" /></label></p>
      <p><label>Checkbox <input type="checkbox" name="check" value="c1" />
        <input type="checkbox" name="check" value="c2" /></label></p>
      <p><label for="rad">Radio</label>
        <input type="radio" name="rad" value="r1" />
        <input type="radio" name="rad" value="r2" /></p>
      <p><label>File <input type="file" name="file" /></label></p>
      <p><label>Select <select name="select1">
        <option value="1">s1</option>
        <option value="2">s2</option>
      </select></label><br />
      <label>Textarea <textarea rows="3" cols="20" name="area"></textarea></label></p>
      <p><input type="submit" /> <input type="reset" /></p>
    </form>
  </body>
</html>
```

Attributi di form

Raggruppa tutti i controlli che debbono raccogliere dati per un destinatario.

Attributi:

- **action**: l'URI dell'applicazione server-side che riceverà i dati
- **method**: il metodo HTTP che deve essere usato per i dati

GET

i dati vengono trasferiti *in chiaro* nella stringa dell'URI (con un punto interrogativo (“?”) come separatore) e quindi sono *visibili*

POST

i dati sono codificati nel corpo del form (metodo *consigliato*)

- **id**: un identificatore univoco per il form
- **enctype**: il metodo di codifica dei dati da usare
 - *default 'application/x-www-form-urlencoded': i codici non alfanumerici sono sostituiti da '%HH' (HH: codice esadecimale del carattere), gli spazi sono sostituiti da '+', i nomi dei controlli sono separati da '&', e il valore è separato dal nome da '=')*
 - *'multipart/form-data' per inviare anche file*

Elementi di form: input

Raccoglie la maggior parte dei tipi di controllo disponibili in un form. Attributi:

- *type*: rappresenta il tipo di controllo da istanziare: text, password, checkbox, radio, submit, reset, file, hidden, image, button
- *id*: l'identificativo del campo
- *value*: il valore di default (per i controlli di testo) o selezionato (per i controlli di tipo bottone) che viene presentato all'utente e passato all'applicazione server-side
- *size* e *maxlength*: dimensione prevista e lunghezza massima
- *disabled*, *checked*: per i bottoni, radio e checkbox

Altri elementi di form: textarea, select, optgroup, option

- *textarea*

Area di inserimento testo. Attributi (oltre a NAME):

- *rows*: il numero di righe previste
- *cols*: il numero di colonne previste

- *select*, *optgroup*

Una lista di opzioni e sotto-opzioni. Attributi (oltre a NAME):

- *size*: il numero di opzioni visibili
- *multiple*: Vero se la lista permette selezioni multiple

- *option*

Una opzione in un *select* o in un *optgroup*

- *selected*: vero se l'elemento è selezionato all'avvio
- *label*: scritta visibile all'utente
- *value*: valore passato all'applicazione server-side

tabindex

Il normale ordine di scorrimento con il tasto tabulatore può essere modificato sulla base dell'importanza dei vari componenti della pagina web:

```
<a href="" tabindex="2">two</a>
...
<input type="text" name="nome"
      title="indica il nome" tabindex="3">
<a href="" tabindex="1">one</a>
```

JavaScript: validare form client-side

un [esempio](#)

In caso di informazioni che richiedono un formato particolare, come le date, fornire un esempio:

```
<label for="data">Data (gg-mm-aaaa)</label>  
<input type="text" name="data" id="data" />
```

dalle Tecniche di Scripting per le Linee guida per l'accessibilità dei contenuti Web (WCAG) 2.0

Nel corso CSS vedremo form più efficaci e ulteriori tecniche di accessibilità

Esercizio

Scrivere (aggiungere al documento dell'esercizio precedente)
un documento XHTML1.1 che contenga elementi meta, paragrafi, intestazioni, attributi, liste e un form

Utilizzate Quanta plus

Contenuto

- [Introduzione](#)
- [Struttura di base](#)
- [Liste](#)
- [Link](#)
- [Altri elementi block-level](#)
- [Altri elementi inline](#)
- [Immagini](#)
- [Tabelle](#)
- [Moduli \(forms\)](#)
- **[Head del documento](#)**

Sezione Meta

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN" "http://www.w3.org/Markup/DTD/xhtml-rdfa-1.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:dc="http://purl.org/dc/elements/1.1/"
      xmlns:foaf="http://xmlns.com/foaf/0.1/"
      xmlns:doap="http://usefulinc.com/ns/doap#"
      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
      xml:lang="it">
  <head>
    <meta name="Description" content="Home page Persona" />
    <meta name="Keywords" content="Semantic Web, Sistemi di Supporto alle Decisioni, Tecnologie Web, Multimedia,
      Web Services, Java Enterprise, W3C, OWL, RDF, XML, XSL, AIDA, CITEL, DTT, CIE" />
    <meta name="Author" content="email1@server.di.posta" />
    <meta name="Author" content="email2@server.di.posta" />
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta name="Language" content="it" />
    <title>Persona :: Home Page</title>
    <link href="stile.css" rel="stylesheet" type="text/css" />
    <link href="stile_spec.css" rel="stylesheet" type="text/css" />
    <link rel="meta" type="application/rdf+xml" title="FOAF" href="./persona.rdf" />
    <script type="text/javascript" src="script.js"></script>
    <script type="text/javascript" src="twitter-html/script.js"></script>
    <style type="text/css">
      .red { color:red; }
    </style>
  </head>
  <body>
    ...
  </body>
</html>
```

Alcuni strumenti

- [W3C validator](#)
- [Tidy](#)
- [Online Tidy](#)
- Quanta plus
- Firefox plugin: [Firebug](#) , [WebDeveloper](#)
- [HTML Kit](#)
- ...
- Aptana
- Bluefish

Grazie per l' attenzione

Domande?

... e [risposte](#)

Se non è sul Web non esiste ...

... troverete sul sito dell' Ufficio (<http://www.w3c.it/>)
le [slide](http://www.w3c.it/education/2012/upra/html/) (<http://www.w3c.it/education/2012/upra/html/>)

Queste slide fanno parte del materiale predisposto per il corso [Architettura del Web](#)