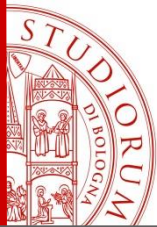




HTML5

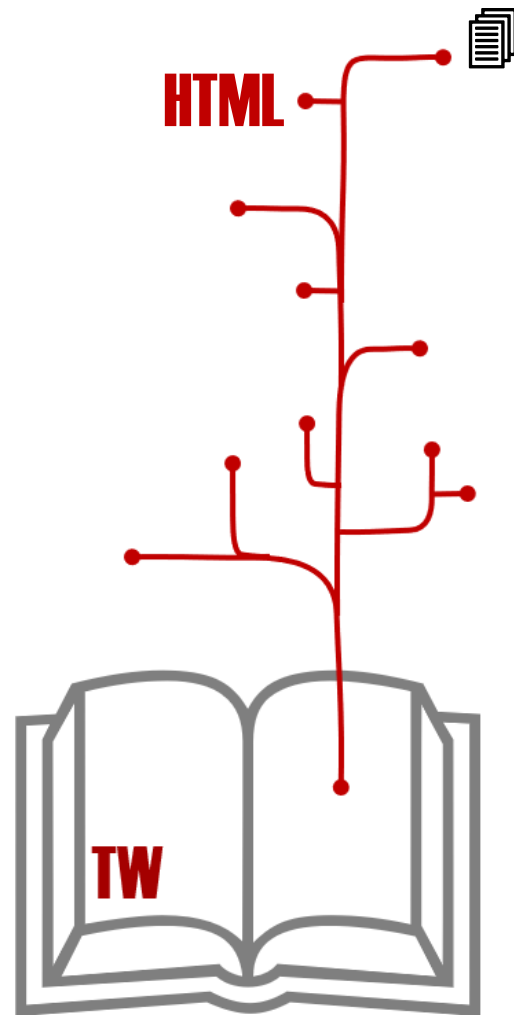
seconda parte

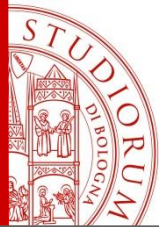




Argomenti

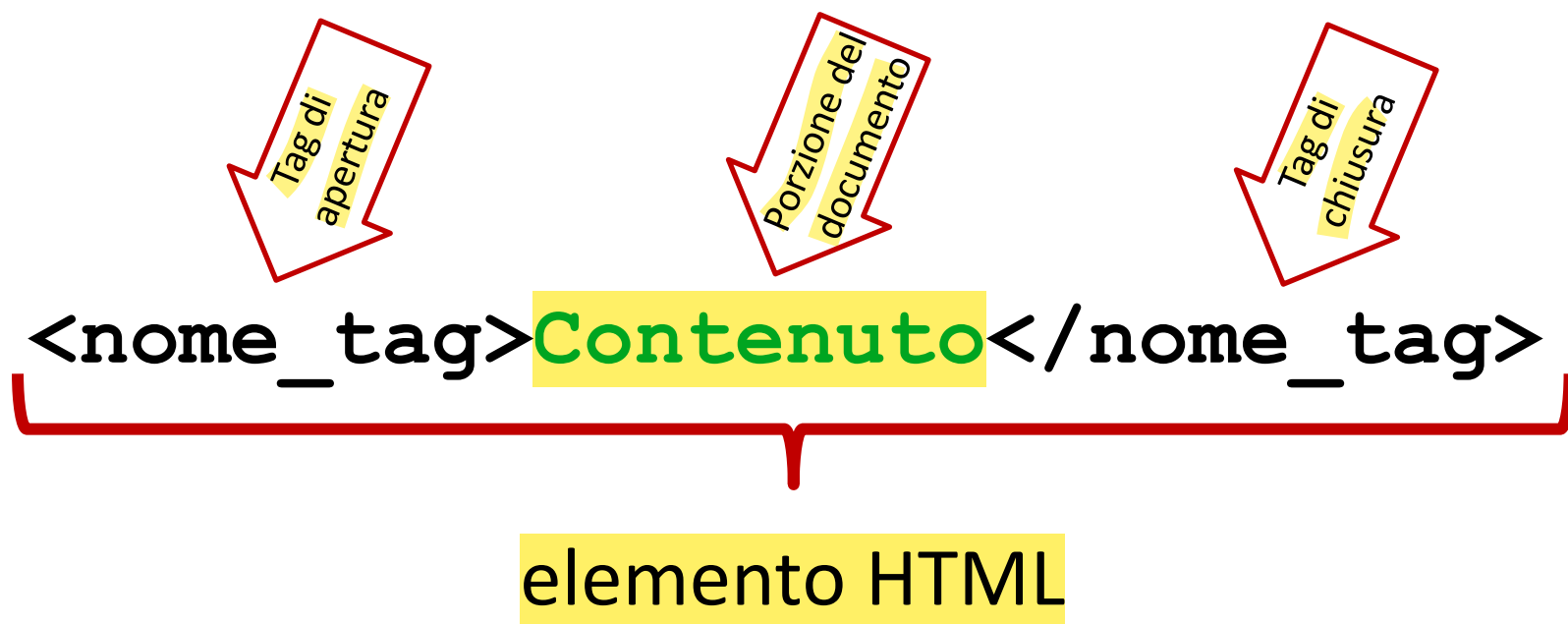
- HTML:
 - Elementi:
 - Sectioning
 - Phrasing
 - Embedded

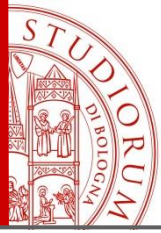




Elementi HTML

- Un elemento HTML è definito da un tag di apertura, un contenuto e un tag di chiusura





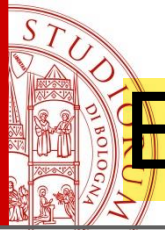
Tag e Attributi

- I tag sono il markup che aggiungiamo al contenuto per dare struttura, enfasi, per definire il ruolo che tale contenuto ricopre all'interno del documento Web (Es: `<p>`, `<h1>`, `<table>`)
- I tag possono essere corredati di uno o più **attributi**, che servono per meglio specificare la funzione o la tipologia dell'elemento, per memorizzare dati o per arricchire di significato il contenuto



Attributi

- Sono coppie *nome-valore* separate dal carattere “=”
- I valori sono racchiusi tra virgolette “”
- Si scrivono lasciando uno spazio dopo il nome del tag di apertura (o dopo lasciando uno spazio dopo le virgolette di chiusura del valore del precedente attributo per lo stesso tag)
- Esempio:
`Università di Bologna`

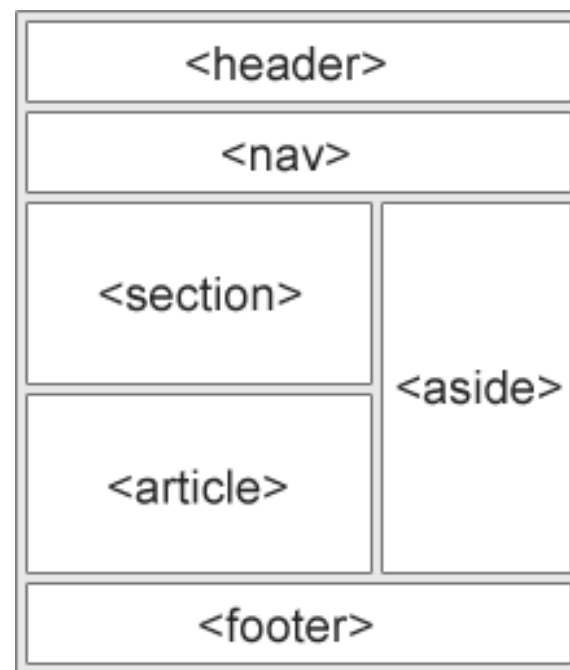


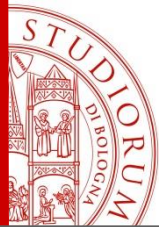
Elementi di blocco e Elementi inline

- **Elementi di Blocco:** il loro comportamento di default nella finestra del browser è quello di essere preceduti e seguiti da una andata a capo, sono nativamente rappresentati come un box (esempi: tabelle, liste, heading, form, paragrafi)
- **Elementi inline:** sono contenuti in un elemento di blocco e non ne intaccano il flusso, ovvero non implicano l'andata a capo, né prima né dopo (esempi: link ipertestuali, elementi per enfatizzare il testo, citazioni)

- Gli elementi della categoria **Sectioning** hanno **funzione strutturale**, ovvero **dividono la pagina in parti con semantica (ruolo) diverso a seconda dell'elemento usato**:

- `<article>`
- `<aside>`
- `<figcaption>`
- `<figure>`
- `<footer>`
- `<header>`
- `<nav>`
- `<section>`

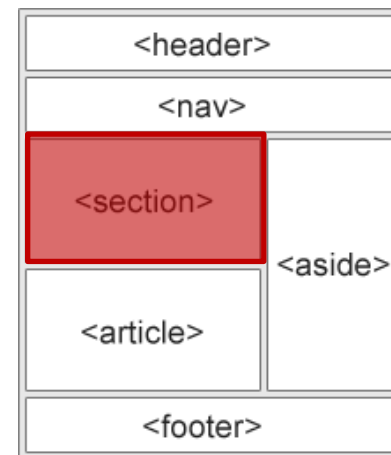




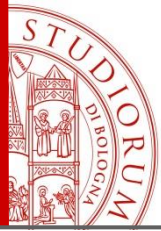
<section>

Un blocco di contenuto che tratta uno specifico argomento o tema del documento, ma che non é indipendente dal documento

- Definisce una **sezione** del documento.
- Nella recommendation HTML5 del W3C: "*A section is a thematic grouping of content, typically with a heading*" (Un raggruppamento tematico di contenuto, solitamente con un'intestazione)
- Una home page potrebbe essere suddivisa in 3 sezioni:
 - una per l'introduzione,
 - una per il contenuto vero e proprio e
 - una per le informazioni sui contatti



http://www.w3schools.com/html/tryit.asp?filename=tryhtml5_section

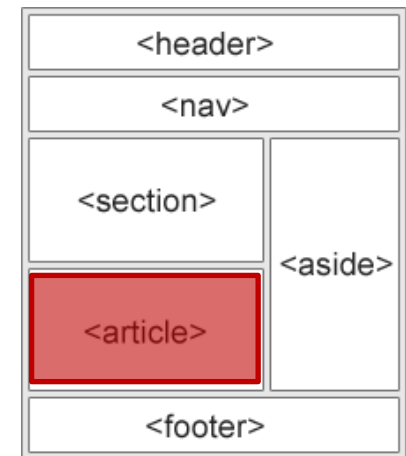


<section> organizza il contenuto della pagina in blocchi tematici, mentre <article> rappresenta contenuti indipendenti e autosufficienti.

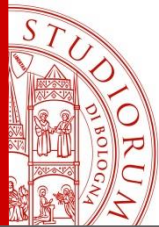
Ogni <article> dovrebbe avere un proprio significato indipendente e contenere tutte le informazioni necessarie affinché il lettore comprenda il contenuto senza dover fare riferimento ad altre parti della pagina.

<article>

- Definisce informazioni indipendenti e auto-contenute:
 - Un **articolo** dovrebbe essere un elemento con un suo senso proprio, che potrebbe essere letto in modo indipendente dal resto della pagina Web.
- Esempi di elementi che potrebbero essere **<article>**:
 - Post su un blog, un forum o un social network
 - Articolo di un quotidiano online

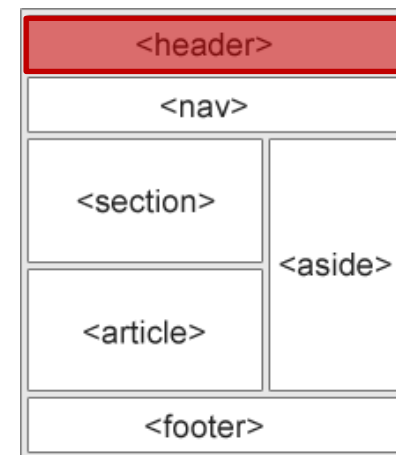


http://www.w3schools.com/html/tryit.asp?filename=tryhtml5_article



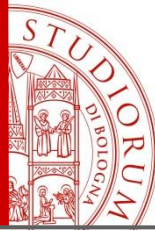
<header>

- Definisce l'intestazione di un documento o di una sua sezione
- Può essere usato come contenitore di un contenuto di tipo introduttivo
- Possono essere presenti più **<header>** in ogni pagina Web



`<header>` in HTML è utilizzato per definire l'intestazione di una pagina o di una specifica sezione della pagina, fornendo una sorta di "introduzione" o di "contenitore informativo" per il contenuto che segue. Se il contenuto introduttivo di una sezione è più complesso di un titolo, il `<header>` è una scelta preferibile, se no utilizza `<h1>...<h6>`

http://www.w3schools.com/html/tryit.asp?filename=tryhtml5_header

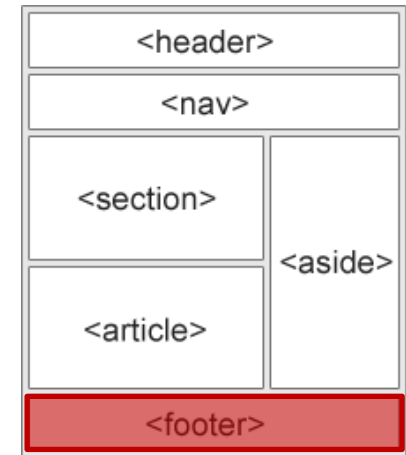


Mentre `<header>` rappresenta l'introduzione, il `<footer>` è una chiusura, solitamente con informazioni complementari o di supporto, che servono a completare la pagina o sezione.

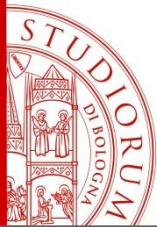
`<footer>`

Il tag `<footer>` in HTML è utilizzato per definire la parte conclusiva di una pagina web o di una specifica sezione. Questo tag è destinato a contenere informazioni di chiusura e dettagli che non fanno parte del contenuto principale, ma che completano il documento o sezione a cui appartengono.

- Definisce il **footer** di un documento o di una sua sezione
- Solitamente un footer contiene le informazioni sull'autore del documento, le informazioni sul copyright, link ai termini d'uso, informazioni sui contatti, ecc
- Possono essere presenti più **<footer>** in ogni pagina Web



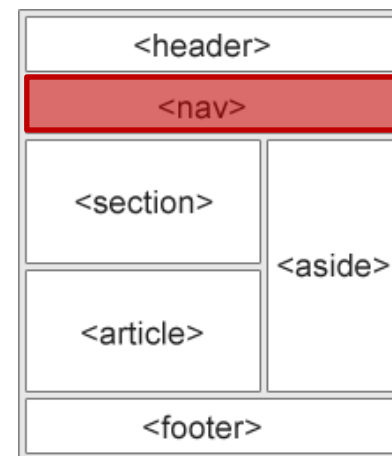
http://www.w3schools.com/html/tryit.asp?filename=tryhtml5_footer



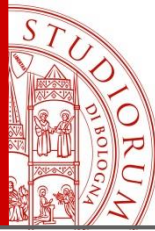
<nav>

- Definisce un insieme di link di **navigazione** (menù o toolbar o altri set di link)

```
<!DOCTYPE html>
<html>
...
<body>
  <nav>
    <a href="/html/">HTML</a> |
    <a href="/css/">CSS</a> |
    <a href="/js/">JavaScript</a> |
    <a href="/jquery/">jQuery</a>
  </nav>
</body>
</html>
```



http://www.w3schools.com/html/tryit.asp?filename=tryhtml5_nav

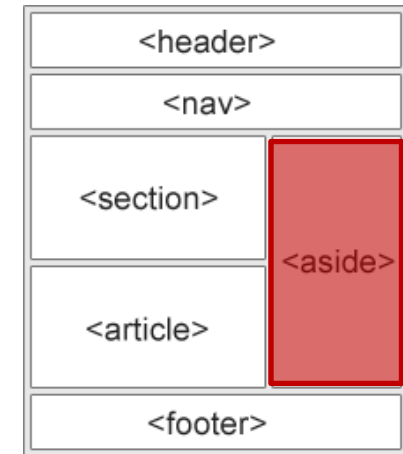


In breve, `<aside>` è ideale per contenuti di contorno che arricchiscono o approfondiscono il tema principale, ma che non devono distrarre il lettore dal testo centrale.

`<aside>`

Il tag `<aside>` in HTML è utilizzato per contenuti che sono correlati al contenuto principale di una pagina o sezione, ma che non fanno parte del flusso principale del testo. È pensato per contenuti collaterali, ovvero informazioni aggiuntive, citazioni, pubblicità o altre note di contesto, che possono arricchire l'esperienza dell'utente senza interrompere il filo della lettura.

- Definisce un **contenuto a latere** rispetto a quelli principali (ma comunque correlati).
- Può essere utilizzato per contenere i contenuti di una barra laterale (sidebar), ma anche per contenuti (separati da quello principale) che sono collaterali, ma non si posizionano necessariamente a lato (per esempio citazioni o banner pubblicitari).



http://www.w3schools.com/html/tryit.asp?filename=tryhtml5_aside

- Gli heading introducono i **titoli** delle diverse sezioni del documento:
 - Gli heading vanno da `<h1>` a `<h6>` a seconda della loro rilevanza (rank)
 - **h1** è quello con maggiore rank e deve rappresentare il titolo principale della sezione.
 - I titoli di rank inferiore devono intestare sottosezioni.
- Sezioni e heading si possono far corrispondere in molti modi diversi, ma noi scegliamo quello che meglio rispetta i principi definiti all'inizio, ovvero:
 - usare sezioni esplicite (sempre con elemento di sezione e titolo, mai solo il titolo)
 - Far combaciare il grado dell'intestazione con il livello di nidificazione previsto.

```
<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>
```



<h1><h2><h3><h4><h5><h6>

- Un esempio, con <h1> e <h2>:



```
<section>
```

```
<h1>Linguaggi di Markup</h1>
```

```
<p>L'insegnamento di Tecnologie Web introduce  
diversi linguaggi di markup tra cui....</p>
```

```
...
```

```
<section>
```

```
<h2>XML</h2> h2 perché fa parte di una section che è sottosezione di un'altra section
```

```
<p>(Extensible Markup Language) è un meta-  
linguaggio di markup, progettato per lo  
scambio e la interusabilità di documenti  
strutturati su Internet. </p>
```

```
</section>
```

```
</section>
```

DOMANDA 1:

Quale struttura descrive correttamente una sezione intitolata «Principale» con due sottosezioni intitolate «Uno» e «Due»:

☐ `<section>
 <h1> Principale </h1>
 <section> <h1> Uno </h1> </section>
 <section> <h1> Due </h1> </section>
</section>`

☒ `<section>
 <h1> Principale </h1>
 <section> <h2> Uno </h2> </section>
 <section> <h2> Due </h2> </section>
</section>`

Prima domanda (segue)



DOMANDA 1:

Quale struttura descrive correttamente una sezione intitolata «Principale» con due sottosezioni intitolate «Uno» e «Due»:

- ☐

```
<section headers = "Principale" >  
  <section headers = "Uno" ></section>  
  <section headers = "Due" ></section>  
</section>
```
- ☐

```
<h1> Principale <section>  
  <h2> <section> Uno </section> </h2>  
  <h2> <section> Due </section> </h2>  
</section></h1>
```

- Nella categoria **Phrasing** rientra il contenuto che rappresenta il testo del documento.
- In particolare introduciamo gli elementi:
 - `<p>`
 - `
`
 - `<div>`
 - ``
 - `<main>`
 - `<a>`
 - Altri elementi che definiscono il ruolo del testo come: `<blockquote>`, `<q>`, `<cite>`, `<i>`, ``, ``, ``, `<small>`, `<abbr>`, `<code>`, `<dfn>`, `<var>`, `<sub>`, `<sup>`, `<ins>`, ``

<p>

- L'elemento inserisce un **paragrafo testuale**.

- **L'andata a capo non basta!**

In HTML, andare a capo nel testo senza utilizzare un tag come <p> non crea un paragrafo con la stessa semantica e non ha la stessa formattazione.

- • Va utilizzato solo quando non esiste un elemento più specifico o semanticamente più idoneo per descrivere quel testo.

- Esempio:

```
<section>
```

```
  <p> Primo paragrafo di testo.</p>
```

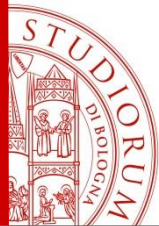
```
  <p> Secondo paragrafo di testo.</p>
```

```
  <p> Terzo paragrafo di testo.</p>
```

```
</section>
```


- L'elemento **
** rappresenta una **interruzione di linea (line break)**:
 - È un elemento vuoto, va scritto con /
 - Deve essere usato solo per interruzioni che sono effettivamente parte del contenuto, come negli indirizzi, nelle poesie o nel codice, **NON** per ottenere effetti grafici.
- Esempio

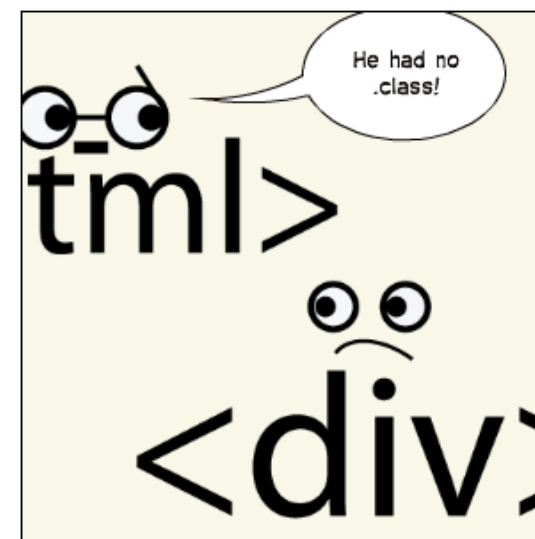
```
<p>Nel mezzo del cammin di nostra vita<br/>mi ritrovai per una selva oscura<br/>ché la diritta via era smarrita.</p>
```

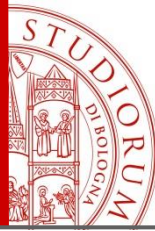


<div>

L'elemento <div> in HTML è un contenitore generico utilizzato per raggruppare altri elementi, soprattutto quando non c'è un tag HTML più specifico o semantico che descriva quel gruppo di elementi.

- L'elemento **<div>** (elemento di blocco) non ha alcun significato proprio ma ha lo scopo di **rappresentare gli elementi in esso annidati e specificare per loro gli attributi class, lang e title.**
 - Viene usato soprattutto per definire l'attributo **class**, ovvero dare a un gruppo di elementi consecutivi uno stesso **stile di presentazione.**
 - La caratteristica presentazionale non deve avere una connotazione semantica (per esempio di sezione) perché in questo caso al posto di **div** dovrebbe essere usato un elemento con semantica (nell'esempio **<section>**).





`` viene utilizzato per raggruppare o stilizzare porzioni di testo all'interno di un flusso di contenuto senza creare interruzioni nel layout. `<div>` Viene utilizzato per raggruppare altri elementi HTML (come testi, immagini, moduli, ecc.) creando una sezione distinta nella pagina.

``

Il tag `` in HTML è un elemento di blocco inline che viene utilizzato per raggruppare e stilizzare porzioni di testo senza apportare significato semantico al contenuto. È simile all'elemento `<div>`, ma mentre `<div>` si occupa di contenuti a livello di blocco, `` è utilizzato per porzioni di testo più piccole e inline.

- L'elemento `` opera in modo simile all'elemento `<div>` ma a livello di testo (è un elemento inline).
 - Non ha alcun significato proprio ma ha lo scopo di **rappresentare il testo in esso contenuto e specificare per esso gli attributi** `class`, `lang` e `title`.
 - Viene usato soprattutto per definire l'attributo **`class`**, ovvero dare a una porzione di testo consecutivo uno stesso **stile di presentazione**.
 - La caratteristica prestazionale, non deve avere una connotazione semantica (per esempio di enfasi) perché in questo caso al posto di **`span`** dovrebbe essere usato un elemento con semantica (nell'esempio ``)

Seconda domanda

**BONUS**

DOMANDA 2:

Per inserire in una sezione l'indirizzo a lato, è opportuno utilizzare:

- ☐ delle andate a capo
- ☐ dei paragrafi con `<p></p>`
- ☐ delle andate a capo con `
`
- ☐ una immagine

Silvia Mirri

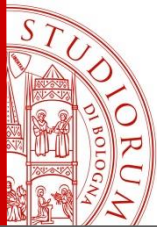
Alma Mater Studiorum – Università di Bologna
Campus di Cesena

Via dell'Università 50 – Cesena (FC)

Tel. +39 0547 338892

<main>

- l'elemento **<main>** raggruppa gli elementi di struttura (come **<section>** o **<article>**) che rappresentano il contenuto principale del documento:
 - il contenuto deve essere caratterizzante di quel documento, quindi vanno esclusi i contenuti che sono ripetuti in diverse pagine (come per esempio le barre di navigazione).
 - ogni documento deve avere un solo <main>.



<i> e non forniscono alcun significato semantico al testo che racchiudono, quindi non utilizzarli

Ruolo del testo

- Elementi che attribuiscono **ruoli** al testo:

MAI
USARLO

– <i>, **testo in voce alternativa**: termini tecnici, frasi idiomatiche, pensieri, testo in altra lingua.

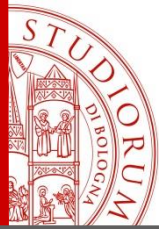
– , **stress emphasis**: un testo o una frase che si pronuncia in modo differente dal resto. Usato per enfatizzare il testo

– , **strong importance**: testo importante

MAI
USARLO

– , **offset text (conventionally styled in bold)**: testo più visibile.

– <small>, **note a margine**. Riduce la dimensione del testo per note o informazioni secondarie.



Ruolo del testo

- altri:

- **<abbr>**, **abbreviazioni o acronimi**. L'attributo **title** è usato per inserire la versione espansa del termine.
- **<var>**, **variabili o costanti** usate in documenti a carattere scientifico (matematico, informatico o altro).
- **<dfn>**, **definizione di un termine**:

← rappresenta la definizione esplicita del termine.

- se ha un attributo title, questo assume il valore di definizione oppure
- se non ha un attributo title, deve contenere solo un elemento **<abbr>** che abbia un **title** che rappresenta la definizione, oppure

← il testo all'interno di <dfn> stesso può costituire la definizione.

- Il testo contenuto in **<dfn></dfn>** è la definizione

- **<code>**, **code, porzioni di codice** (anche HTML, ovviamente)

← Se <dfn> non ha un attributo title, può contenere solo un elemento <abbr> che ha un attributo title per fornire la definizione.

- Esistono altri elementi di questo tipo, che trovate nelle specifiche.



<sub> e <sup>

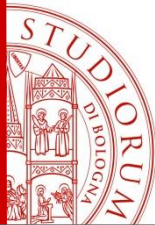
- <sub> e <sup> corrispondono ad apice e pedice.
- ESEMPIO:

```
<p>This text contains  
  <sub>subscript</sub> text.</p>
```

```
<p>This text contains  
  <sup>superscript</sup> text.</p>
```
- http://www.w3schools.com/tags/tryit.asp?filename=tryhtml_sup

This text contains subscript text.

This text contains superscript text.



L'elemento `<blockquote>` è utilizzato per citare un brano di testo proveniente da una fonte esterna. È comunemente usato per estratti lunghi, come discorsi, articoli o altre forme di contenuti.

Citazioni

- Ci sono vari elementi per inserire **citazioni**:
 - `<blockquote>` per parti di contenuto che vengono **citare** da una sorgente esterna (specificabile attraverso l'attributo opzionale `cite`).
http://www.w3schools.com/tags/tryit.asp?filename=tryhtml_blockquote_test
 - `<q>` simile a `blockquote` ma agisce su un **breve testo**. Si può usare l'attributo `cite`.
http://www.w3schools.com/tags/tryit.asp?filename=tryhtml_q
 - `<cite>` per citare i riferimenti ad un **lavoro creativo** (come una bibliografia). Deve includere il titolo del lavoro, o il nome dell'autore o l'URL di riferimento
http://www.w3schools.com/tags/tag_cite.asp



<ins> e

- Gli elementi **<ins>** e **** servono a **marcare le modifiche al documento**. In particolare:
 - **<ins>** si usa quando si aggiunge contenuto al documento.
 - **** si usa quando si cancella contenuto dal documento.
- Per entrambi gli attributi sono:
 - **cite** per specificare fonti aggiuntive.
 - **datetime** per specificare data e ora della modifica.
- ESEMPIO:

```
<p>My favorite color is <del>blue</del>  
  <ins>red</ins>!</p>
```
- http://www.w3schools.com/tags/tryit.asp?filename=tryhtml_del

My favorite color is ~~blue~~ red!

- Il contenuto **Embedded** ha lo scopo di importare risorse o contenuto dentro il documento
- Gli embedded fanno riferimento all'area dei sistemi multimediali
- Alcuni elementi embedded prevedono un contenuto **fallback**, che viene usato quando la risorsa esterna non può essere utilizzata (e.g. se ha un formato non supportato).
- Vediamo:
 - ``
 - `<audio>`
 - `<video>`
 - `<embed>`
 - `<object>`
 - `<canvas>`



<figure><figcaption>

- <figcaption> definisce la **didascalia** di una immagine
- <figure> raggruppa l'immagine e la sua didascalia
- Esempio



<figure>

<figcaption>Fig.1 - Pulpit Rock,
Norvegia.</figcaption>

</figure>

https://www.w3schools.com/tags/tryit.asp?filename=tryhtml_figcaption

- Le **immagini inline** sono definite attraverso l'elemento ****.
- Gli attributi obbligatori sono:
 - **src**: specifica l'URL del **file contenente l'immagine**. L'URL deve fare riferimento a una "*non-interactive, optionally animated, image resource that is neither paged nor scripted*". Sono consentiti i formati PNG, GIF, JPEG e alcuni tipi (non paginati, non scriptati) di PDF, XML, SVG.
 - **alt**: **testo alternativo**, viene visualizzato in caso il browser non riesca a mostrare l'immagine e in caso di immagini disabilitate. È indispensabile agli utenti non vedenti (ne riparleremo nella lezione sull'accessibilità).

Sul tag metti usemap="#nomeMappa".

Nel documento definisci un elemento <map name="nomeMappa"> contenente uno o più <area> (shape + coords + href).

Quando l'utente clicca, il browser decide localmente quale link seguire, in base alle coordinate e alle aree definite.

- **Attributi opzionali sono:**
 - **usemap**, specifica che l'immagine è una **mappa lato client**
 - **ismap**, specifica che l'immagine è una mappa lato server
 - **width**, specifica la larghezza dell'immagine in pixel
 - **height**, specifica l'altezza dell'immagine in pixel
 - **longdesc**, URL che porta alla **descrizione lunga per l'accessibilità**

- **Esempio:**

```

```

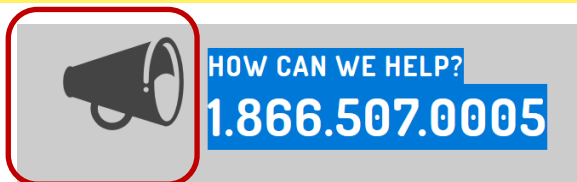


http://www.w3schools.com/tags/tryit.asp?filename=tryhtml_image_test

Un ragionamento su img

- Facciamo un primo ragionamento sulle immagini.
- Gli attributi **src** e **alt** sono OBBLIGATORI
- Ma l'alternativa serve sempre? (SI)

– ?



```

```

– ?



```

```

– ?



SONO LINK!

```
<a href="...URL..."></a>
```

– ?

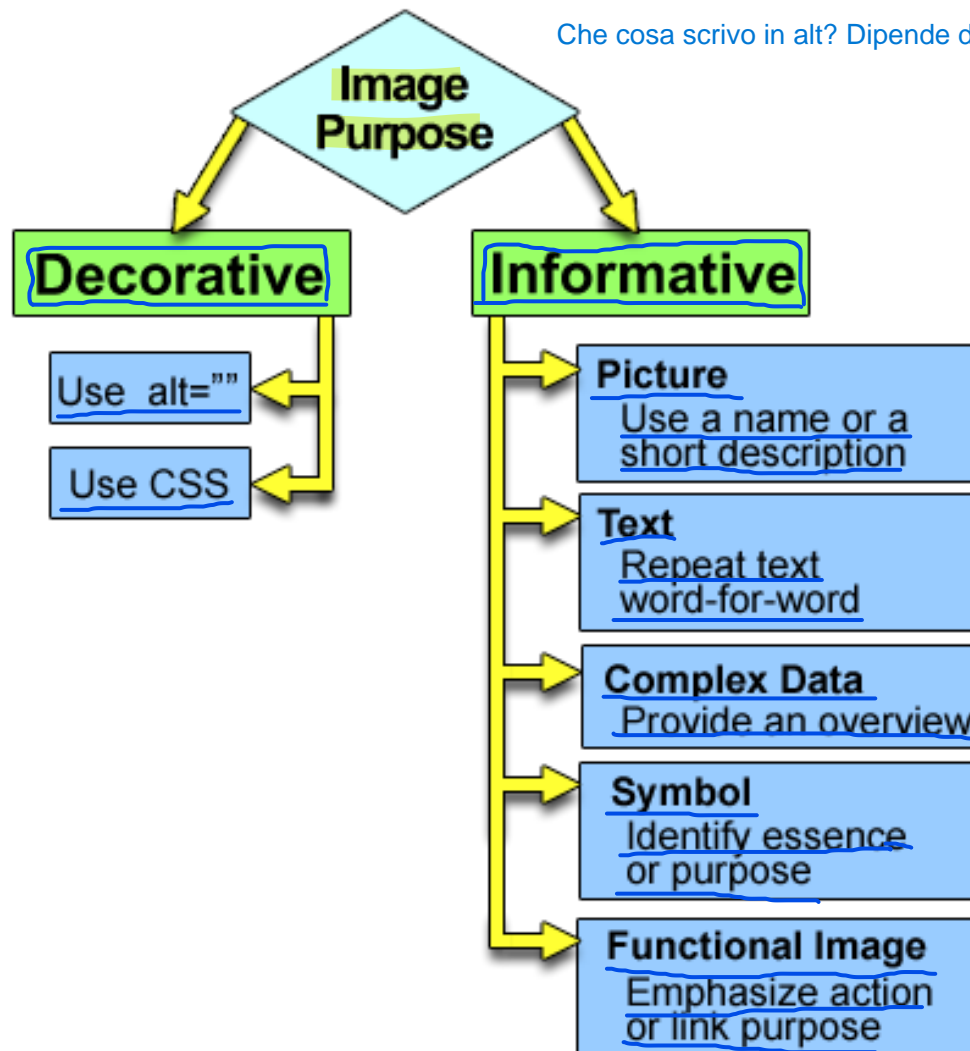


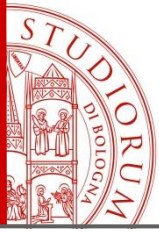
```

```

Una prima indicazione

Che cosa scrivo in alt? Dipende dal tipo di immagine...





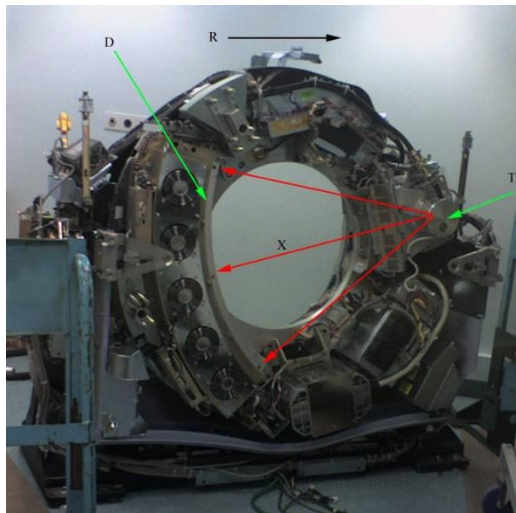
Alternative testuali

- Se l'immagine è complessa, va descritta in modo dettagliato e verboso.
- In HTML 4.01 veniva utilizzato l'attributo **longdesc**, che permetteva di collegare all'immagine un file separato con la descrizione estesa. Questo attributo è stato deprecato con HTML5, ma vi è in corso una diatriba molto accesa per poterlo mantenere
- **longdesc** non è supportato da tutti i browser ma può esserlo dagli screenreader
- In assenza di altre soluzioni fattibili, efficaci e condivise dalla comunità, nell'ambito di questo corso manteniamo il suo utilizzo

Alternative testuali

```

```



figcaption

CT scanner with cover removed to show internal components.
Legend: T: X-ray tube
D: X-ray detectors
X: X-ray beam
R: Gantry rotation

A modern (2006) CT scanner with the cover removed, demonstrating the principle of operation. The X-ray tube (T) and the detectors (D) are mounted on a ring shaped gantry. The patient lies in the center of the gantry while the gantry rotates around them. This arrangement, a broad fan-shaped X-ray beam (X) with rotating source and detectors, is the 'third-generation' configuration. This scanner is capable of helical scanning - the gantry is able to rotate freely (R) while the patient moves continuously through the plane of the scan.

longdesc

Terza domanda

BONUS

DOMANDA 3:

Quale alt indichereste per questa immagine (fonte Repubblica, occhiello a destra):

- ☐ "Champions League"
- ☐ "Il Napoli si rilancia"
- ☐ "Superato il Feyenoord"
- ☐ ""

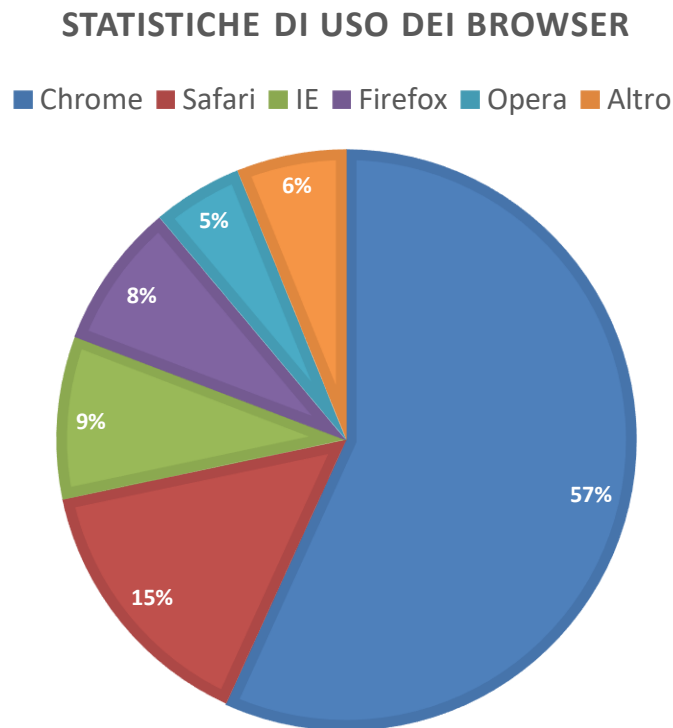
<a>



Esempio

- **ESERCIZIO 1 ESAME**

Scrivere il codice di un documento HTML il cui body contiene solo una sezione intitolata «Browser» con due sotto sezioni: la prima si intitola «Client Web» e include un paragrafo di testo.; la seconda è intitolata «Statistiche» include la figura a lato.





Soluzione

```
<!DOCTYPE html>
<html lang="it">
<head><title>Titolo del
documento</title></head>
<body>
<section>
  <h1>Browser</h1>
  <section>
    <h2> Client Web </h2>
    <p> paragrafo di testo </p>
  </section>
...

```




Soluzione

```
...  
<section>  
  <h2> Statistiche </h2>  
  <figure>  
      
    <figcaption>  
      Statistiche di uso dei browser  
    </figcaption>  
  </figure>  
</section>  
</section>  
</body>  
</html>
```



Soluzione: longdesc

```
<!DOCTYPE html>
<html lang="it">
<head><title>Descrizione pie chart</title></head>
<body>
  <p> Uso dei Browser in percentuale: Chrome 56,8%;
    Safari 14,9%; IE 9,1%; Firefox 8,1%; Opera 5%;
    Altro 6,1%
  </p>
</body>
</html>
```

Nota: andava bene anche un semplice file di testo

<video>

- L'elemento **<video>** definisce un modo standard per **includere un video** in una pagina Web
- Prima di HTML5, non esisteva uno standard per mostrare i video nelle pagine Web, che potevano essere mandati in play solo con un plug-in (come Flash):

Attributi

- **controls** aggiunge i controlli per il video: play, pausa, volume, ecc
- **width** definisce la larghezza del video; se non specificato il valore di **height** (altezza), viene calcolato mantenendo le proporzioni originali del video
- Il testo contenuto in **<video></video>** viene mostrato nel caso in cui il browser non supporti l'elemento



<video>

```
<!DOCTYPE html>
<html lang="it">
<head>
  <title>Monsters and Co</title>
  <meta charset="UTF-8"/>
  <style>
    video {
      position: fixed; top: 100px; left: 100px;
    }
  </style>
</head>
<body>
  <video width="700px" controls>
    <source src="monster.mp4" type="video/mp4"/>
    <source src="monster.ogg" type="video/ogg"/>
    Il tuo browser non supporta il tag video
  </video>
</body>
</html>
```

<video>

- L'attributo **controls** aggiunge i **controlli** per il video: play, pausa, volume, ecc
- L'attributo **width** definisce la larghezza del video; se non specificato il valore di **height** (altezza), viene calcolato mantenendo le proporzioni originali del video
- È possibile proporre il video in **diversi formati**, usando gli elementi **<source>**. Il browser utilizzerà il primo riconosciuto e supportato. Tra i valori ammessi per l'attributo **src** del tag **<source>** **NON** ci sono i link ai video di YouTube
- Il testo prima del tag di chiusura **</video>** viene mostrato nel **caso in cui il browser non supporti l'elemento <video>**

<video>

- Per far partire automaticamente un video è possibile usare l'attributo **autoplay**

```
<video width="700px" autoplay>
```

```
...
```

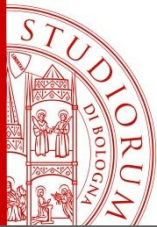
```
</video>
```

- E' possibile usare contemporaneamente gli attributi **controls** e **autoplay**

```
<video width="700px" autoplay controls>
```

```
...
```

```
</video>
```



<audio>

- L'elemento <audio> definisce un modo standard per includere un audio in una pagina Web
- Prima di HTML5, non esisteva uno standard per includere gli audio nelle pagine Web, che potevano essere mandati in play solo con un plug-in (come Flash)

```
<audio controls>
```

```
  <source src="loversroad.mp3"  
    type="audio/mpeg"/>
```

```
  Il tuo browser non supporta il tag audio
```

```
</audio>
```

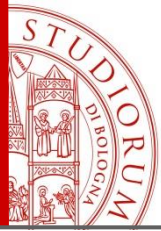


<audio>

```
<!DOCTYPE html>
<html>
<head>
  <title>Lover's Road</title>
  <meta charset="UTF-8"/>
</head>
<body>
  <audio controls>
    <source src="loversroad.mp3"
      type="audio/mpeg"/>
    Il tuo browser non supporta il tag audio
  </audio>
</body>
</html>
```

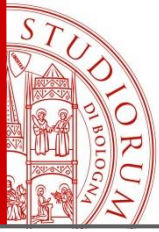

<audio>

- L'attributo **controls** aggiunge i **controlli** per l'audio: play, pausa, volume, ecc
- Il testo prima del tag di chiusura **</audio>** viene mostrato nel caso in cui il browser non supporti l'elemento **<audio>**
- È possibile proporre l'audio in diversi formati, usando gli elementi **<source>**. Il browser utilizzerà il primo riconosciuto e supportato



<object>

- Supportato da tutti i browser
- Definisce un oggetto incluso in un documento HTML
- Usato per includere audio, video, animazioni e plug-in (come applet Java, lettori PDF, player Flash) nelle pagine Web
- In modo simile a <source>, ammette versioni alternative dell'oggetto (con <object> annidati). Il primo formato riconosciuto e supportato è quello mandato in playout dal browser

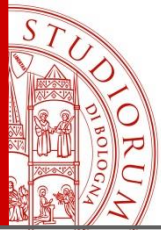


<object>

```
<!DOCTYPE html>
<html>
<head>
    <title>Animazione Flash</title>
    <meta charset="UTF-8"/>
</head>

<body>
    <object width="400px" height="300px"
        data="audio1.swf">
    </object>
</body>

</html>
```



<object>

- Può essere utilizzato anche per includere un documento HTML in un HTML:

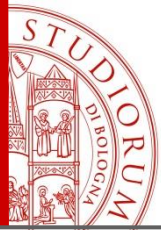
<body>

```
<object width="800px" height="600px"  
data="prova.html">
```

```
</object>
```

</body>

http://www.w3schools.com/html/tryit.asp?filename=tryhtml_object_html



<embed/>

- Supportato dalla maggior parte dei browser, è un elemento vuoto (non ha chiusura)
- Definisce un oggetto incluso in un documento HTML, come **<object>**
 - È stato supportato a lungo dai browser Web, anche se non è mai stato parte di uno standard HTML, prima di HTML5
 - Contrariamente a **<object>** non ammette alternative di alcun tipo

http://www.w3schools.com/tags/tryit.asp?filename=tryhtml5_embed



<embed/>

```
<!DOCTYPE html>
<html>
<head>
    <title>Animazione Flash</title>
    <meta charset="UTF-8"/>
</head>

<body>
    <embed width="400px" height="300px"
        src="audio1.swf"/>
</body>

</html>
```



<embed/>

- Può essere utilizzato anche per includere un documento HTML in un HTML (embed2.html):

```
<body>
```

```
<embed width="800px" height="600px"  
src="prova.html" />
```

```
</body>
```

https://www.w3schools.com/tags/tryit.asp?filename=tryhtml5_embed_html

<canvas>

- Una importante innovazione di HTML5 è la **possibilità di disegnare direttamente sulla pagina**
- L'elemento **<canvas>** fornisce le API necessarie per la generazione e il rendering dinamico di grafica, diagrammi, immagini e animazioni.
- L'elemento **<canvas>** definisce un'area rettangolare in cui disegnare direttamente immagini bidimensionali e **modificarle in relazione a eventi, tramite funzioni Javascript.**



<canvas>

- La larghezza e l'altezza del canvas sono specificati tramite gli attributi **width** e **height** dell'elemento **<canvas>**.
- Le coordinate (0,0) corrispondono all'angolo in alto a sinistra.
- Gli oggetti non sono disegnati direttamente sul canvas ma all'interno del contesto, recuperato tramite un metodo Javascript dell'elemento **<canvas>** chiamato **getContext()**:
 - Questo metodo è parte di una vasta libreria utile per disegnare figure, colorarle, trasformarle, etc.
 - Altra tecnologia per la grafica vettoriale e l'animazione prevista da HTML5 è quella basata su SVG
 - Entrare nei dettagli del funzionamento di **<canvas>** e di SVG non è negli obiettivi di questo corso (siamo nell'area dei sistemi multimediali)
- **Esempi:** <http://corehtml5canvas.com/code-live/>

Mappe

- Una **mappa** è un'immagine in cui alcune aree sono interattive, attivano un link o altre azioni.
- Una mappa può essere realizzata:
 - **client-side**: il browser esamina la locazione del click e attiva/gestisce l'interazione.
 - **server-side**: il server esamina la locazione del click e attiva/gestisce l'interazione.
- Esistono molti tipi di mappe che servono a fornire servizi avanzati di ricerca e localizzazione.
- In HTML esiste anche un semplice tipo di mappe lato client che può essere realizzato attraverso markup.

<map>

- Trattiamo velocemente questi elementi sapendo che NON sono quelli alla base di sistemi di mappe che usiamo solitamente (basati su Javascript e tecnologie server side).
- **<map>** è classificato come elementi phrasing.
- Per realizzare una mappa con **<map>**:
 - Usare **usemap**, specificando l'id del tag di riferimento.
 - Definire una mappa con **<map>**.
 - Specificate aree sensibili, forma, dimensioni e link da attivare con il tag **<area>**.

```
<area shape="circle" coords="90,58,3"  
      href="mercur.htm" alt="Mercury"/>
```

<map>

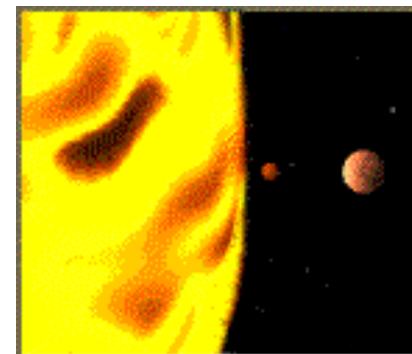
Usiamo un'immagine a cui collegarci una mappa. Su questa immagine, definiamo delle aree dove, andandoci a cliccare, ci reindirizza ad una pagina html

- Esempio:

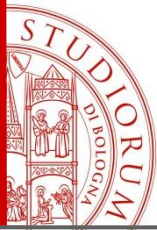
```

```

```
<map name="planetmap">
  <area shape="rect" coords="0,0,82,126"
    href="sun.htm" alt="Sun"/>
  <area shape="circle" coords="90,58,3"
    href="mercur.htm" alt="Mercury"/>
  <area shape="circle" coords="124,58,8"
    href="venus.htm" alt="Venus"/>
</map>
```



http://www.w3schools.com/tags/tryit.asp?filename=tryhtml_areamap



Riferimenti

- Vedi piattaforma
- Standard W3C:
<https://html.spec.whatwg.org/multipage/>
- Living Standard:
<http://www.w3.org/TR/html5/dom.html#kinds-of-content>

