

HTML

L'HTML è un linguaggio studiato ed implementato per essere utilizzato nel **World Wide Web**.

I documenti HTML rappresentano pagine ipermediali contenenti:

- Testo
- Immagini
- Suoni
- Collegamenti ipertestuali ad altre pagine (link)
- Programmi

Le pagine vengono ospitate su server web e visualizzate mediante dei programmi client i **browser**.

HTML è l'acronimo di **Hyper Text Markup Language** un documento html altro non è che un file di testo scritto rispettando una determinata sintassi che consente di contrassegnare e disporre i vari elementi all'interno di una pagina; le varie indicazioni sono date mediante appositi marcatori che sono detti **tag**.

Nel corso degli anni il linguaggio HTML si è evoluto così si sono venute ad avere diverse versioni quindi è stato creato un organismo il W3C (World Wide Web Consortium, www.w3c.org) che definisce gli standard di questo linguaggio a livello mondiale.

Vediamo adesso alcuni elementi fondamentali del linguaggio HTML.

Il tag **<!doctype>** è l'intestazione della pagina e definisce il tipo di documento scritto (HMTL, XHTML, HTML5, ...) specifica quindi le regole della versione HTML che si sta utilizzando nella pagina ed inoltre contiene l'indicazione al file delle regole che sta sul sito del W3C questo tag è opzionale.

Il tag **<html>** è l'elemento radice della pagina ed al suo interno sono contenuti tutti gli altri tag della pagina.

Nel tag **<head>** viene specificato il titolo e le caratteristiche della pagina html.

Il tag **<body>** è l'elemento nel quale viene inserito tutto il contenuto della pagina html.

Il tag **<body>** e gli altri tag relativi ai contenuti posseggono diversi attributi che consentono di impostare l'aspetto della pagina.

I tag presenti nel <body> si possono suddividere in tre categorie:

- block level tag
- text level tag
- empty tag

I **block level tag** sono elementi che formano blocchi di testo e che determinano l'andata a capo dopo il tag di chiusura e possono contenere sia elementi **block level** che **text level**.

I **text level tag** sono elementi che formattano solo il testo e non determinano l'andata a capo dopo il tag di chiusura e possono contenere solo elementi **text level**.

Gli **empty level tag** sono elementi vuoti e possono servire a vari scopi.

I colori in HTML possono essere impostati per nome (in inglese o utilizzando il formato RGB con 8 bit (256 livelli) per il rosso 8 bit (256 livelli) per il verde ed 8 bit (256 livelli) per il blu.

I colori in formato RGB vengono impostati utilizzando numeri in formato esadecimale quindi si usano due cifre per ciascuno dei tre colori.

I paragrafi

I tag <p> e <div> sono dei block level tag ed in genere racchiudono un paragrafo la differenza tra i due tag è che se si usa il tag <p> vengono inserite due righe vuote una prima del tag ed una dopo il tag.

Altri tag che consentono di andare a capo sono gli empty level tag
 che va semplicemente a capo ed <hr /> che crea una linea di separazione.

Elementi di formattazione del testo

Vediamo adesso quali sono i principali tag che consentono la formattazione del testo:

- tag il testo racchiuso in questo tag è visualizzato nella pagina html in grassetto (**bold text**)
- tag il testo racchiuso in questo tag è visualizzato nella pagina html come importante (**important text**)

- tag **<i>** il testo racchiuso in questo tag è visualizzato in corsivo nella pagina html (**italic text**)
- tag **** il testo racchiuso in questo tag è visualizzato enfatizzato nella pagina html (**emphasized text**)
- tag **<mark>** il testo racchiuso in questo tag è visualizzato evidenziato nella pagina html (**emphasized text**)
- tag **** il testo racchiuso in questo tag è visualizzato barrato nella pagina html (**deleted text**)
- tag **<ins>** il testo racchiuso in questo tag è visualizzato sottolineato nella pagina html (**inserted text**)
- tag **<sub>** il testo racchiuso in questo tag è visualizzato come pedice nella pagina html (**subscript text**)
- tag **<sup>** il testo racchiuso in questo tag è visualizzato apice nella pagina html (**superscript text**)

Lo stile

A praticamente tutti i tag html può essere associato uno stile che modifichi l'aspetto del tag questa operazione avviene aggiungendo al tag l'attributo **style**.

Esempio 1:

```
<div style = "font-family: verdana" >Testo scritto col font verdana</div>
```

```
<div style = "color: red; font-family: courier" >Testo di colore rosso</div>
```

L'attributo style consente di impostare un insieme molto vario di proprietà elenchiamone alcune delle più comuni:

Proprietà	Effetto
background-color	Consente di impostare il colore di sfondo
color	Consente di impostare il colore
font-family	Consente di impostare il tipo di carattere (verdana, ariel, courier, ...)
font-size	Consente di impostare le dimensioni del carattere
text-align	Consente di impostare l'allineamento

Ogni tag HTML ha le proprie proprietà che consentono di modificare lo stile.

Liste

Le liste si possono avere due tipi di liste numerate e non.

Le liste non numerate si ottengono col tag `<ul>` che racchiude gli elementi della lista che devono essere all'interno del tag `<li>`.

Esempio 2:

```
<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

Le liste numerate si ottengono col tag `<ol>` che racchiude gli elementi della lista che devono essere all'interno del tag `<li>`.

Esempio 3:

```
<ol>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

Un ulteriore tipo di lista è una lista con descrizione si ottiene col tag `<dl>` che racchiude gli elementi della lista che devono essere all'interno del tag `<dt>` mentre le loro descrizioni sono contenute dal tag `<dd>`.

Esempio 4:

```
<dl>
  <dt>Coffee</dt>
  <dd>- black hot drink</dd>
  <dt>Milk</dt>
  <dd>- white cold drink</dd>
</dl>
```

Le liste possono essere modificate variandone lo stile utilizzando l'attributo `style`.

Tabelle

Le tabelle sono realizzate utilizzando il tag `<table>` che racchiude le righe realizzate col tag `<tr>` che a loro volta racchiudono le colonne realizzate col tag `<td>`. Per le colonne della prima riga di una tabella si utilizza in genere il tag `<th>` che sta ad indicare le intestazioni della tabella.

Esempio 5:

```
<table >
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

Importante per le colonne (tag **<td>**) è l'attributo **colspan** che consente di mantenere allineate le colonne tra più righe quando tra le righe vi è un diverso numero di colonne.

Esempio 6:

```
<table>
  <tr>
    <th>Opera</th>
    <th>Nome</th>
    <th>Cognome</th>
    <th>Anno</th>
  </tr>
  <tr>
    <td>Gioconda</td>
    <td>Filippo</td>
    <td>Neri</td>
    <td>1987</td>
  </tr>
```

```

<tr>
  <td>Le camelie</td>
  <td colspan="2">Sconosciuto</td>
  <td>1587</td>
</tr>
<tr>
  <td>I girasoli</td>
  <td>Fabio</td>
  <td>Bruni</td>
  <td>2001</td>
</tr>
<tr>
  <td>Il mulino</td>
  <td colspan="3">Sconosciuti</td>
</tr>
</table>

```

Le tabelle possono essere modificate variandone lo stile utilizzando l'attributo **style** per gli esempi 5 e 6 si vuole usare lo stile:

```

table {
  width:50%;
  border: 1px solid black;
  font-size: 200%;
}
td {
  border: 1px solid black;
}
th {
  border: 1px solid black;
  background: #B0B0B0;
  bold;
}

```

Immagini

Vediamo adesso come si possono introdurre delle immagini nelle pagine html.

Per inserire una immagine si usa il tag vuoto **** con questo tag è indispensabile utilizzare degli attributi che consentono la gestione delle immagini i più importanti sono:

- **src** individua il file contenente l'immagine che si vuol visualizzare
- **alt** contiene il testo da visualizzare in alternativa all'immagine

Ovviamente si può utilizzare l'attributo style per modificare alcune caratteristiche dell'immagine.

Esempio 7:

```

```

Link o collegamenti ipertestuali

I link generalmente sono utilizzati per consentire la navigazione tra le pagine web ma si possono anche utilizzare per gestire o scaricare dei contenuti.

Il tag che si utilizza per creare dei link è **<a>** si tratta di un tag non vuoto.

L'attributo più importante di questo tag è **href** che contiene l'indirizzo della pagina oppure del file da scaricare.

Un altro attributo importante è l'attributo target che può valere:

- **_self** (default): apre il documento nella stessa finestra/tab del link
- **_blank** – apre il documento in una nuova finestra/tab
- **_parent** – apre il documento nel frame contenente il frame del link
- **_top** – apre il documento nell'intero corpo della finestra

Inoltre si possono utilizzare i link anche con immagini al posto del testo.

Vediamo adesso degli esempi di utilizzo dei link:

Esempio 8:

```
<a href="https://chironi.edu.it/index.php" >Visita il sito della  
scuola ITC G.Chironi</a>
```

```
<a href="https://chironi.edu.it/index.php" >  
  
</a>
```

```
  
  
<map name="workmap">  
  <area shape="rect" coords="34,44,270,350" alt="Computer"  
href="computer.htm" />  
  <area shape="rect" coords="290,172,333,250" alt="Phone"  
href="phone.htm" />  
  <area shape="circle" coords="337,300,44" alt="Coffee"  
href="coffee.htm" />  
</map>
```

Il tag form

Il tag form consente l'invio dei dati ad una altra pagina. In genere l'invio dei dati è avviene da una pagina principale che contiene i vari campi dei form e consente all'utente di effettuare delle scelte, scrivere del testo, inserire dei dati in generale ad una pagina richiamata dalla principale e che effettua "il lavoro" vero e proprio di processare e raccogliere i dati.

Normalmente la pagina che riceve i dati è una pagina con codice sorgente di un linguaggio che lavora lato server.

Vedremo adesso la sola pagina che riguarda la form.

Il tag **<form>** è un elemento di blocco, come il `<p>`, quindi il tag `<form>` lascia uno spazio prima dell'apertura e dopo la chiusura.

Esempio:

```
<form name="dati" action="risposta.php">  
...  
</form>
```

L'attributo **name** serve per indicare il nome della form mentre l'attributo **action** individua l'URL della pagina che processerà i dati.

Quando creiamo una form possiamo scegliere due metodi di invio: GET e POST.

Il metodo GET fa sì che la pagina di risposta viene contattata ed i dati vengono inviati in un unico step. Nell'URL della pagina di risposta potremo allora vedere tutti i parametri nella barra degli indirizzi.

Esempio:

risposta.php?nome=Gianni&cognome=Cecchi

Spesso tuttavia i server hanno delle limitazioni per quel che riguarda il metodo GET e non consentono di inviare form con valori superiori a 255 caratteri complessivi.

Il metodo GET è dunque particolarmente indicato per form con pochi campi e pochi dati da inviare e che non riguardino dati sensibili.

La sintassi per l'invio in get è:

```
<form name="dati" action="risposta.php" method="GET">
```

L'invio dei dati usando il metodo POST invece avviene in due parti

- 1) viene contattata la pagina sul server che deve processare i dati
- 2) vengono inviati i dati stessi.

In questo modo i parametri non vengono visualizzati e non ci sono limiti sulla lunghezza dei caratteri.

La sintassi per il metodo POST è:

```
<form name="dati" action="risposta.php" method="POST">
```