**HTML PARTE 1**

Che cos’è l’HTML?

HTML (**H**yper**t**ext **M**arkup **L**anguage) è il codice che viene utilizzato per strutturare una pagina web e i suoi contenuti.

Ad esempio il contenuto potrebbe essere strutturato all’interno di un insieme di paragrafi, o in una lista a pallini, oppure presentato attraverso immagini o tabelle di dati.

HTML fornisce gli strumenti per strutturare una contenuto che vogliamo presentare nel nostro sito web.

E’ definito come linguaggio di markup ( di marcatura) e consiste in una serie di elementi che si usano per inglobare diverse parti del contenuto in modo tale da farlo apparire in un determinato modo o da farlo comportare in un determinato modo.

Il conenuto è inserito nei cosiddetti tag che sono delle etichette che creano gli elementi.

I tag permettono di creare paragrafi, link ipertestuali, mostrare del testo in grassetto, in corsivo, di un certo colore e via così.

Vediamo un esempio:

Ho il seguente testo

My cat is very grumpy

(grumpy = irritabile, scontroso)

Voglio che questa riga appaia in un paragrafo singolo

<p>My cat is very grumpy</p>

**Vediamo l’anatomia di un elemento HTML**



1. Il tag di apertura ( nel nostro caso <p>) consiste nel nome dell’elemento racchiuso tra parentesi angolari

Tale tag stabilisce dove comincia l’elemento o da dove inizia ad aver effetto.

1. Il tag di chiusura ( </p>) dice dove finisce l’elemento o dove finisce il suo effetto .

(Uno degli errori tipici di chi scrive pagine HTML è dimenticarsi di chiudere un tag aperto. Questa dimenticanza può portare a strani comportamenti della pagina web)

1. Il contenuto è ciò che vogliamo venga manipolato dai tag. In questo caso è solo testo.
2. Tutti i componenti nel loro insieme formano **l’elemento**

Gli elementi possono anche avere **attributi**, come nel seguente caso:



Gli attributi contengono delle informazioni supplementari sull’elemento , che non vuoi che appaiano nel contenuto.

Qui **class** è il nome dell’attributo e **editor-note** è il valore dell’attributo

L’attributo class permette di dare all’elemento un identificativo che può essere utilizzato in seguito per individuare l’elemento con informazioni di stile o altre cose.

Le caratteristiche di un attributo devono essere le seguenti:

* Un attributo deve avere uno spazio tra esso e l’elemento a cui si applica ( o l’attributo precedente se l’elemento ha più di un attributo
* Il nome dell’attributo seguito dal segno =
* Il valore dell’attributo tra apici

**Nota**: Valori di attribute semplici che non contengano spazi (o uno dei seguenti caratteri  " ' ` = < > ) possono rimanere senza apici, ma si raccomanda di quotarli tutti, dato che rende il codice più leggibile e consistente.

Gli elementi possono essere innestati (ovvero l’uno dentro l’altro)

<p>My cat is <strong>very</strong> grumpy.</p>

Qui la parola very viene enfatizzata.

Alcuni elementi inoltre possono essere vuoti, ovvero non prevedono contenuto.

Un esempi tipico è il tag <img>

<img src="images/firefox-icon.png" alt="My test image">

Ci sono due attributi , ma non c’è il tag di chiusura </img> e nessun contenuto.

Questo perché un’immagine non prevede un contenuto per essere visualizzata.

Attributi booleani

A volte si incontreranno attributi senza valore. Questi sono noti come **attributi booleani**, e possono avere solo un valore, che è generalmente lo stesso del nome dell’attributo.

Come esempio vediamo l’attributo disabled che si può assegnare ad elementi input per disabilitarli (visivamente diventano grigi), in modo tale che l’utente non possa digitare al loro interno alcun valore.

<input type="text" disabled="disabled">

Come scorciatoia, si può scrivere nel seguente modo (aggiungiamo un campo di input non disabilitato per vedere la differenza. Da provare!)

<input type="text" disabled>

<input type="text">

Omettere apici attorno agli attributi

A volte si possono incontrate degli elementi che contengono attributi senza apici. Questa modalità funziona in alcuni casi , ma in altri può dare problemi nella struttura dell’html.

Vediamo un esempio con un link

<a href=https://www.mozilla.org/>favorite website</a>

Questo element funzione, ma appena aggiungiamo il titolo non funziona più:

<a href=https://www.mozilla.org/ title=The Mozilla homepage>favorite website</a>

Il browser non intrerpeta correttamente l’attributo titolo , e pensa che sia in realtà tre attributi:

1. Un attributo titolo con valore **The**
2. Un attributo booleano di nome **Mozilla**
3. Un attributo boolean di nome **homepage**

Pertanto il consiglio è quello di utilizzare sempre gli apici per evitare questo tipo di problemi e per rendere il codice più leggibile.

Apici singoli o doppi?

<a href="http://www.example.com">A link to my example.</a>

<a href='http://www.example.com'>A link to my example.</a>

Questi due elementi sono entrambi validi e la scelta dipende da una questione di stile.

Bisogna però fare attenzione a non mescolare apici singoli e doppi. Il seguente esempio è errato:

<a href="http://www.example.com'>A link to my example.</a>

Se si utilizza uno dei due tipi di apici , si può includere l’altro tipo all’interno dei valori dell’attributo:

<a href="http://www.example.com" title="Isn't this fun?">A link to my example.</a>

Ma se si vuole includere come valore di un attributo un apice già utilizzato per racchiudere il contenuto, allora bisogna ricorrere alle **entità HTML**

Per esempio il seguente codice non funziona

<a href='http://www.example.com' title='Isn't this fun?'>A link to my example.</a>

Pertanto dobbiamo scriverlo nel seguente modo

<a href='http://www.example.com' title='Isn&#39;t this fun?'>A link to my example.</a>

**Riferimenti a entità HTML**

In HTML i caratteri  <, >,",' e  & sono caratteri speciali. Sono parte della sintassi HTML , pertanto se si vogliono includere in un testo devono essere interpretati come elementi del testo e non come caratteri speciali. Utilizziamo pertanto al loro posto dei riferimenti, codici speciali che rappresentano caratteri e possono essere utilizzati in queste circostanze. Ciascun riferimento a carattere inizia per & e termina con un ;

|  |  |
| --- | --- |
| Carattere letterale | Riferimento a carattere equivalente |
| < | &lt; |
| > | &gt; |
| " | &quot; |
| ' | &apos; |
| & | &amp; |

Vediamo un esempio

<p>In HTML, you define a paragraph using the <p> element.</p>

<p>In HTML, you define a paragraph using the &lt;p&gt; element.</p>

Block vs Inline elements

Ci sono due importanti categorie di elementi in HTML:

Block e inline

* Gli elementi a livello di blocco formano un blocco visibile sulla pagina. Appariranno su una nuova riga da qualsivoglia elemento che li precede. Anche qualsivoglia elemento che li segue apparirà su una nuova riga. Gli elementi a livello di blocco sono elementi di tipo strutturale sulla pagina e rappresentano paragrafi, liste, menu di naviagazione, footer…

Un elemento di blocco non può essere innestato in un elemento inline, ma può esserlo in altri elementi di blocco.

* Gli elementi inline sono elementi contenuti in elementi di blocco e circondano solo piccole parti del contenuto html, non interi paragrafi o raggruppamenti di contenuti. Quando si applica un elemento inline questo non apparirà su una nuova riga : appaiono all’interno di paragrafi di testo, ad esempio come link (<a>) oppure enfatizzano una o più parole, come nel caso di <em> e <strong>

Vediamo il seguente esempio:

<em>first</em><em>second</em><em>third</em>

<p>fourth</p><p>fifth</p><p>sixth</p>

<em> è un elemento inline pertanto i tre elementi sono uno vicino all’altro.

<p> è invece un paragrafo, un elemento a livello di blocco. Pertnato fourth, fifth e sixth appaiono su righe differenti.

**Nota**: I termini “block” e “inline”, usati in questo contesto, non devono essere confuse con I tipi di box CSS , che hanno lo stesso nome. Sono correlate per default, ma se si cambiano le proprietà visuali del CSS, non si cambia di sicuro la categoria di elemento.

**Note**: Pagine di riferimento con una lista di elementi blocco e inline: [Block-level elements](https://developer.mozilla.org/en-US/docs/Web/HTML/Block-level_elements) e [Inline elements](https://developer.mozilla.org/en-US/docs/Web/HTML/Inline_elements).