

Fondamentalmente l' XHTML o eXtensible HTML, è un linguaggio basato sull'ormai tradizionale HTML, con la caratteristica di essere portabile su tutte le piattaforme, a partire dai tradizionali browser Web fino a dispositivi come cellulari, palmari, PDA, ecc... L'ultima release di HTML, la 4.01, risale ormai al 1997 e da allora non è stata più aggiornata. Quello che è stato fatto con XHTML non è risultato un aggiornamento del linguaggio verso una nuova versione, la 5 per intenderci, ma una razionalizzazione onde evitare da qui in poi gli errori dovuti alla permissività nella elaborazione del codice. La sintassi di molte pagine HTML è infatti sporca e piena di errori, che i browser lasciano passare con non curanza, oltre al fatto, che fino ad oggi non è esistita una modalità specifica (standard) per costruire le pagine. XHTML, tenta proprio di scardinare questa brutta abitudine, attraverso le rigide specifiche di XML. L' XML esiste dal 1998 ed è una sorta di "super-linguaggio" o Meta-Linguaggio che consente la creazione di nuovi linguaggi di marcatura. Oggi è alla base di tutte le nuove specifiche tecnologiche rilasciate dal W3C e adottate ormai come standard dall'industria informatica. I principali obiettivi di XML, sono, la facilità di creazione dei documenti, il supporto di più applicazioni, la chiarezza e la comprensibilità. Ecco dunque che la seguente espressione riassume bene l'oggetto di questa discussione > **HTML + XML = XHTML** <, in cui nulla viene sostituito da nulla, ma si ha l'integrazione di due diverse filosofie di sviluppo, quella permissiva e quella restrittiva. Ciò ha portato a 3 diverse versioni del nuovo linguaggio : l' XHTML 1.0, XHTML BASIC e XHTML 1.1.

L' XHTML 1.0 è la versione pubblicata il 26 gennaio 2000 e seguita da una versione rivista dell'ottobre 2001. Consiste, come detto, nella riscrittura in XML di HTML 4.0 e si basa su tre DTD (Document Type Definition) :

- DTD Strict
- DTD Transitional
- DTD Frameset

L' XHTML Basic E' una versione ridotta del precedente linguaggio, pensata per dispositivi mobili (PDA, cellulari), contenendo solo gli elementi che si adattano a questi dispositivi (esclude, ad esempio, i frames che non hanno ovviamente senso in tale contesto). E' destinata a sostituire WML come linguaggio di base per le applicazioni WAP.

In ultimo l' XHTML 1.1 basato sulla DTD Strict della versione 1.0 in cui l'insieme dei tag che definiscono la struttura di un documento, sono raggruppati in una serie di moduli indipendenti, che possono essere implementati o esclusi secondo le necessità. Secondo il W3C è la base della futura estensione di XHTML con altri set di linguaggi o moduli, anche personalizzati.

Iniziamo col delineare cosa è una DTD, che consiste in pratica in un modello che definisce il tipo di XHTML usato. In altre parole ogni pagina XHTML deve far riferimento ad una delle tre DTD, senza la quale non è in grado di determinare quali sono gli elementi accettati in una pagina web "TAGS", con la conseguenza che non può essere determinato se tale pagina può essere o no valida.

La DTD strict è la forma più rigida di definizione dell'XHTML, non permettendo di utilizzare alcun Tag o attributo che specifici degli stili degli elementi della pagina; non possono quindi essere utilizzati colori, font o stilizzazioni CSS direttamente all'interno del documento: il tutto dev'essere definito in un file CSS esterno. Questo modello delinea perfettamente la nuova filosofia XHTML, in cui l'HTML viene utilizzato così come era stato pensato all'inizio, cioè rivolto solo alla definizione della struttura di una pagina web e non alla forma in cui si presenta la medesima (stile e layout). Le responsabilità per uno uso scorretto del HTML ricadono del tutto nei creatori di siti web e non

nelle mani degli sviluppatori del linguaggio che fin dal 1996, rilasciarono attraverso il W3C la versione definitiva di CSS1. L'idea di fondo era chiara su come dovevano essere sviluppate le pagine HTML ma a causa del fatto che per avere browser che supportassero questi standard si è dovuto attendere il 2000-2001, gli sviluppatori di siti web si sono arrangiati, con ad esempio un uso indiscriminato delle tabelle per la formattazione della pagina, non considerando che esse erano state create solo per la presentazione dei dati. Inserendo tag non ammessi (forse in futuro anche le tabelle), semplicemente il documento (pagina) creato non è valido. Per verificarne che un documento sia valido e ben formato si può ricorrere al sito della W3C <http://validator.w3c.org> in cui è possibile effettuare la validazione di qualunque documento presente in rete, basta inserire l'URL della pagina e cliccare su "Validate this page". Se invece si deve controllare una pagina che non è presente in internet, perché si trova sul nostro computer, in locale, si ha la possibilità di validare i documenti tramite l'upload del file sui server sempre del W3C. Usate quindi l'url <http://validator.w3c.org/file-upload.html>. Se il documento è valido riceverete un messaggio "NO ERRORS FOUND" (figura 1).

No errors found! *



Congratulations, this document validates as XHTML 1.0 Strict!

Nella DTD strict alcuni elementi non sono più supportati : `<applet>`, `<basefont>`, `<center>`, `<dir>`, ``, `<frame>`, `<frameset>`, `<iframe>`, `<isindex>`, `<menu>`, `<noframes>`, `<s>`, `<strike>`, `<u>` , e quindi non possono più essere utilizzati, mentre per i tag rimanenti sono cambiate altre cose come per esempio nel caso degli attributi :

- sono esclusi tutti gli attributi del tag `<body>` tranne quelli comuni
- non si può usare `align` per l'allineamento del testo in paragrafi e altri elementi
- non è supportato l'attributo `target` per i link e i form
- per una tabella (`<table>`) non si possono specificare il bordo, il colore di sfondo (`bgcolor`) o l'allineamento (`align`)
- le celle di tabella (`<td>`) non supportano il colore di sfondo, la larghezza (`width`), l'altezza (`height`). Supportano invece l'allineamento del testo (`align`)

Con la DTD Transitional invece si sono poste le basi per il passaggio, più che mai morbido, dalla vecchia alla nuova filosofia, permettendo a chi volesse convertire il proprio documento da Html a Xhtml di continuare ad utilizzare anche tutti i tags deprecati. Il nome stesso, transitional, ne evidenzia gli scopi. E' quindi la Dtd più "bonaria", che meno si avvicina allo standard Xhtml vero e proprio, ma semplicemente, è quella più facile da usare. Questa versione ammette l'utilizzo di stilizzazioni interne al documento, ma è consigliabile limitarne l'uso al massimo per avere una ampia portabilità del codice verso client differenti dai browser web, come cellulari, palmari, PDA, ecc...

La DTD Framset è identica alla Transitional, ma va usata quando si utilizzano i frame. L'unica differenza è in pratica la sostituzione del tag `<body>` con `<frameset>` nella pagina principale.

Per poter dichiarare una delle tre DTD bisogna aggiungere una stringa all'inizio della pagina che faccia riferimento al tipo di DTD e al luogo ove reperire tale modello :

DTD Strict

```
<!DOCTYPE html  
PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3c.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

DTD Transitional

```
<!DOCTYPE html  
PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3c.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

DTD Frameset

```
<!DOCTYPE html  
PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"  
"http://www.w3c.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
```

In questo caso I modelli si trovano presso il sito del W3C e le pagine con questa dicitura fanno diretto riferimento a tale sito. Pur non essendo ancora diffuso, verrà però un giorno in cui la stragrande maggioranza delle pagine nel web sarà XHTML, a quel punto tale riferimento sarà un po' difficile da raggiungere. Per chi vuol mettere le mani avanti sin da ora è possibile scaricare la definizione delle DTD ed inserirle in locale sul proprio sito. In questo caso si dovrà modificare l'URL nella intestazione come segue :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
http://www.miodominio.it/DTD/xhtml1-strict.dtd>
```

In questo caso si considera che la DTD sia presente nella cartella DTD nel sito **ww.miodominio.it**.

Le dichiarazioni DOCTYPE viste sopra sono formate da due sezioni, nella prima è definito il tipo di modello a cui fa riferimento la pagina, mentre nella seconda vi è l'URL di dove tale modello può essere trovato. La prima sezione si chiama FPI ovvero Identificatore Formale Pubblico. Tale dichiarazione non ha alcun effetto sulla pagina ma è un riferimento che serve al validatore per stabilire quali sono le regole di convalida da utilizzare. La parola chiave PUBBLIC indica che il modello di DTD è pubblico ed è stato rilasciato dal W3C, in alcuni casi può essere utile utilizzare dei modelli personali o privati, in tal caso si usa la parola chiave SYSTEM. Nel caso dell'URL anche se la DTD fosse presente nel proprio sito, è sempre meglio utilizzare l'estensione assoluta e non quella relativa, che potrebbe dare alcuni problemi per la validazione.

Per costruire una pagina valida bisogna seguire delle regole ben precise che con XHTML sono obbligatorie, ad esempio :

- 1) I tags e i loro attributi vanno tutti scritti in minuscolo!
- 2) Tutti i tags devono essere chiusi e chiusi nella stessa maniera!
- 3) Tutti i tags devono essere chiusi nello stesso ordine! (annidati nello stesso ordine)
- 4) Tutti gli attributi devono essere chiusi fra virgolette!
- 5) Ogni attributo deve avere un valore, anche fittizio!
- 6) Per identificare un elemento si deve usare ID e non NAME! (meglio se tutti e due)

7) Nell'uso degli script bisogna far attenzione ai caratteri "Pericolosi". Questi sono caratteri che magari all'interno di una pagina html non creerebbero guai ma con la strettezza del XHTML possono essere mal interpretati. Ad esempio in Javascript > significa "maggiore di ma in XHTML indica la chiusura di un tag. In questo caso è consigliato caricare esternamente file javascript e non includerli direttamente nella pagina. Se ciò non può essere fatto si può ovviare con il costrutto HTML CDATA..... (o semplicemente non validare la pagina).

8) Altri caratteri poi devono essere sostituiti in quanto non riconosciuti, si va dalle vocali accentate fino a caratteri come la &.

| Carattere | Sostituzione |
|-----------|--------------|
| & | & |
| à | á |
| è | é |
| ì | í |
| ò | ó |
| ù | ú |

9) Quando si collegano CCS esterni è opportuno che questi utilizzino le minuscole.

In particolare per quanto riguarda il punto 2 bisogna far notare che devono essere chiusi anche i tag che non avrebbero in teoria una chiusura (TAG VUOTI), ad esempio
 dovrà essere scritto
. I seguenti frammenti di codice esemplificano le norme sopra esposte :

<div> I tags devono essere minuscoli </div>

<p> I tags devono essere chiusi nella stessa maniera </p>

<p>I tags vuoti
 devono essere chiusi</p>

<p> I tags devono essere chiusi nello stesso ordine </p>

<form type= "post" action= "login.php" > (attributi fra virgolette)

<option selected="selected" >test</option> <form action="action" ></form> (valori degli attributi)

A queste regole devono essere poi aggiunte anche quelle del normale codice HTML come ad esempio :

1. un elemento <a> non può contenere altri elementi <a>
2. l'elemento <pre> non può contenere gli elementi , <object>, <big>, <small>, <sub>, <sup>
3. l'elemento <button> non può contenere <input>, <select>, <textarea>, <label>, <button>, <form>, <fieldset>, <iframe>, <isindex>
4. l'elemento <label> non può contenere un altro elemento <label>
5. l'elemento <form> non può contenere l'elemento <form>

In particolare sulle regole che formano la portabilità del linguaggio oggi puntano molte aziende che producono telefoni, pda o simili, come Nokia, Motorola, Ericsson o Siemens che guardano ad XHTML per la piattaforma WAP 2 in cui tutta l'evoluzione dei servizi mobili sarà fondata sull'integrazione tra XHTML e CSS, con il supporto delle necessarie tecnologie sul lato server. Una pagina web si dovrebbe presentare in modo da incorporare diversi CSS per ciascun supporto, identificare il browser e il supporto hardware in modo da vedere su un PC il layout standard e su un cellulare un layout "ridotto" e adatto alle caratteristiche del mezzo, senza che i contenuti cambino. In più la validazione delle pagine dovrebbe rendere le medesime più accessibili rispetto a quelle scritte in HTML, obbligando ad esempio il testo come sostituto alle immagini o la possibilità di poter utilizzare tecnologia vocale.

Ultima caratteristica di XHTML è la struttura del linguaggio che è suddivisa in moduli. I tags sono organizzati in gruppi in modo da costituire blocchi attualmente distinti in due categorie: i moduli fondamentali (**Core Modules**), che rappresentano l'insieme dei tag che devono essere supportati da qualsiasi interprete XHTML, e i moduli di estensione (**Extension Modules**), moduli opzionali il cui supporto può dipendere dal tipo di browser o dispositivo di visualizzazione utilizzato. I moduli fondamentali previsti dall'attuale definizione comprendono:

- **Structure Module**
raggruppa l'insieme dei tag che definiscono la struttura di un documento XHTML (**<html>**, **<head>**, ecc.)
- **Text Module**
raggruppa l'insieme dei tag di qualificazione del testo (**<h1>**, **<h2>**, **<div>**, **<p>**, ****, **<cite>**, ecc.)
- **Hypertext Module**
contiene il tag **<a>**
- **List Module**
contiene i tag che consentono di specificare liste (****, ****, ****, ecc.)

I moduli di estensione sono numerosi e ciascuno contiene i tag relativi alla definizione di particolari elementi di un documento XHTML. Tra i principali ricordiamo:

- **Forms**
raggruppa l'insieme dei tag che consentono di definire le form e i controlli ad esse legate (**<form>**, **<input>**, ecc.)
- **Tables**
contiene i tag per la definizione di tabelle (**<table>**, **<tr>**, **<td>**, ecc.)
- **Images**
contiene il tag ****
- **Frames**
raggruppa l'insieme dei tag che consentono di strutturare un documento in frame (**<frameset>**, **<frame>**, ecc.)
- **Scripting**
contiene i tag che consentono di inserire o meno script (**<script>**, **<noscript>**)

TAGS

Ecco un elenco dei tags in xhtml :

- 1) Il tag **<body>** è utilizzato per includere il corpo del documento.

```
<body>
```

Corpo della pagina

```
</body>
```

- 2) il Tag **<head>** delimita l'intestazione del documento. Fornisce informazioni per la interpretazione del corpo della pagina stessa.

```
<head>
```

Metadati

```
</head>
```

- 3) il Tag **<html>** Serve per racchiudere l'intero documento.

```
<html xmlns="specifica W3C" xml:lang="lingua" lang="lingua per  
retrocompatibilità">
```

Documento

```
</html&>
```

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="it" lang="it">
```

...

```
</html&>
```

- 4) il Tag **<title>** compare tra i tag head del documento. Specifica il titolo del documento stesso.

```
<title>Titolo del documento</title>
```

- 5) il Tag **<abbr>** introduce un'abbreviazione o un acronimo.

```
<abbr>Acronimo o abbreviazione</abbr>
```

- 6) il Tag **<acronym>** ha la stessa funzione di **<abbr>**

```
<acronym>Acronimo o abbreviazione</acronym>
```

7) il Tag **<address>** definisce un indirizzo. Il tag non modifica l'aspetto del testo.

```
<address>Indirizzo</address>
```

```
<address>Via Tagliata, 45 - Genova</address>
```

8) il Tag **<blockquote>** indica che il testo racchiuso è una citazione. I browser visualizzano il testo con un livello di rientro superiore rispetto al resto del paragrafo.

```
<blockquote cite="URL di riferimento" title="testo
informativo">Citazione</blockquote>
```

```
<blockquote cite="http://www.pippo.net">Non siamo materia che aspetta, ma
schemi che si perpetuano.</blockquote>
```

9) il Tag **
** inserisce un ritorno a capo forzato nel testo.

```
Riga di sopra.<br />
```

```
Riga di sotto.
```

10) il Tag **<cite>** ha la funzione di introdurre una citazione.

```
<cite title="testo informativo" lang="lingua">Testo della citazione</cite>
```

```
<cite title="Un noto proverbio" lang="it">Anno bisesto, anno funesto!</cite>
```

11) il Tag **<code>** serve per introdurre il codice sorgente di un programma nel documento.

```
<code title="testo informativo" lang="linguaggio">Codice sorgente</code>
```

```
<code title="Inclusione" lang="php" >include ("file.php");</code>
```

12) il Tag **<dfn>** è una funzione è simile al tag **<dt>**, ma con la differenza che richiama l'attenzione di un utente sul fatto che viene definito un nuovo termine nel discorso.

```
<dfn lang="lingua" title="testo informativo">Definizione</dfn>
```

```
Un quadrato ha quattro <dfn lang="it" title="lati">lati</dfn>.
```

13) il Tag **<div>** ha la funzione di indicare un gruppo di elementi della pagina. L'attributo charset indica il set di caratteri utilizzabile, di default il valore predefinito è ISO8859-1.

```
<div class="classe" charset="set di caratteri" lang="lingua" title="testo informativo">Elemento da raggruppare</div>
```

```
<div lang="it">Elemento da raggruppare</div>
```

14) il Tag **** indica al browser di enfatizzare il testo contenuto. Permette ai dispositivi assistivi di marcare il testo come enfatizzato (ad esempio: i browser vocali).

```
<em>Testo da enfatizzare</em>
```

```
<em>Attenzione!</em> Pagina in costruzione.
```

15) i Tags **<h1>** ... **<h6>** Introducono nella pagina diversi livelli di titoli. Da h1 (primo livello) a h6 (ultimo livello).

```
<h1 lang="lingua" title="testo informativo">Titolo</h1>
```

```
<h1>Titolo della pagina</h1>
```

16) il Tag **<kbd>** ha la funzione di indicare che il testo in oggetto deve essere inserito tramite la tastiera.

```
<kbd>Combinazione di tasti</kbd>
```

```
Premi <kbd>ALT + F1</kbd>
```

17) il Tag **<p>** permette di suddividere un paragrafo.

```
<p lang="lingua" title="testo informativo">Testo del paragrafo</p>
```

```
<p>Questo è un paragrafo</p>
```

18) il Tag **<pre>** identifica una porzione di testo preformattato. Il browser dunque deve rispettare le spaziature e i ritorni a capo presenti nel testo.

```
<pre>Testo preformattato</pre>
```

`<pre>Questo è un testo preformattato</pre>`

19) il Tag `<q>` Introduce una citazione breve.

`<q cite="URI di riferimento" lang="lingua" title="testo informativo">Citazione</q>`

`<q title="Una citazione di prova">Non siamo materia che aspetta, ma schemi che si perpetuano</q>`

20) il Tag `<samp>` è utilizzato per visualizzare un esempio di output di un programma.

`<samp>Output del programma</samp>`

`<samp>Output del programma</samp>`

21) il Tag `` ha la funzione di raggruppare logicamente una porzione di testo.

`Testo`

`Testo`

22) il Tag `` viene utilizzato per porre una forte enfasi su una certa parte di testo.

`Testo in forte enfasi`

`Testo in forte enfasi`

23) il Tag `<var>` introduce una variabile in senso matematico o informatico.

`<var>Variabile</var>`

`<var>Casa</var>`

24) il Tag `<a>` definisce un collegamento all'interno o verso l'esterno di un documento.

`Nome del collegamento`

`pippo.net`

25) il Tag **<dd>** viene utilizzato all'interno di una lista per indicare la definizione della voce introdotta dal tag **<dt>**.

```
<dd lang="lingua" title="titolo informativo">Descrizione dell'elemento</dd>
```

```
<dd lang="it" title="La Cappella Sistina">La Cappella Sistina</dd>
```

26) il Tag **<dl>** delimita una lista di definizioni. L'attributo **compact** riduce la spaziatura tra le righe rispetto allo stile standard.

```
<dl lang="lingua" title="testo informativo" compact="compact">
```

```
...
```

```
</dl>
```

```
<dl lang="it" title="Glossario" compact="compact">
```

```
...
```

```
</dl>
```

27) il Tag **<dt>** introduce un termine in una lista di definizione racchiusa tra i tag **<dl>** ... **</dl>**.

```
<dt class="classe di appartenenza" lang="lingua" title="testo informativo">Termine</dt>
```

```
<dt lang="it" title="World Wide Web Consortium">W3C</dt>
```

28) il Tag **** acronimo di list item, definisce un elemento in una lista (sia ordinata, sia non ordinata).

```
<li lang="lingua" title="titolo informativo" type="circle | disc | square | a | A | i | I | 1" value="numero progressivo">Elemento</li>
```

```
<li type="circle" />Roma</li>
```

29) il Tag **** definisce una lista ordinata.

```
<ol lang="lingua" title="testo informativo"> Definizione della lista </ol>
```

30) il Tag **** viene utilizzato per definire una lista non ordinata di elementi.

```
<ul type="circle | disc | square" compact="compact" lang=lingua">Lista</ul>
```

31) il Tag **<object>** serve per inserire nel documento un oggetto generico.

```
<object standby="Testo per l'attesa del caricamento" data="URI del file"
type="MIME type"> Testo alternativo </object>
```

32) il Tag **<param>** viene utilizzato per passare parametri al tag object.

L'attributo valuetype può assumere i seguenti valori: data (un numero), object (un oggetto) e ref (un URI).

```
<param name="nome del parametro" value="valore del parametro"
valuetype="data | object | ref" type="MIME type" />
```

33) il Tag **** permette di formattare una stringa di caratteri in grassetto (bold).

```
<b>Testo in grassetto</b>
```

34) il Tag **<big>** ingrandisce il testo.

```
<big>Testo</big>
```

35) il Tag **<hr>** permette di inserire una linea orizzontale di separazione nella pagina.

```
<hr />
```

36) il Tag **<i>** Consente di inserire del testo in corsivo.

```
<i lang="lingua" title="testo informativo">Testo in corsivo</i>
```

37) il Tag **<small>** fa sì che il testo interessato sia visualizzato a grandezza inferiore rispetto a quella predefinita.

```
<small>Testo</small>
```

38) il Tag **<sub>** indica che il testo interessato deve essere visualizzato con lo stile pedice o deponente.

```
<sub>Testo</sub>
```

39) il Tag **<sup>** indica che il testo interessato deve essere visualizzato con lo stile apice o esponente.

```
<sup>Testo</sup>
```

40) il Tag **<tt>** consente di inserire un testo con spaziatura fissa.

```
<tt>Testo</tt>
```

41) il Tag **** specifica una porzione di testo cancellata dal documento che tuttavia rimane leggibile nella pagina (testo barrato).

```
<del cite="URL" datetime="data di cancellazione" lang="lingua">Testo  
cancellato</del>
```

42) il Tag **<ins>** ha la funzione di indicare che una porzione di testo è stata inserita in un secondo tempo nel documento.

```
<ins cite="URI di riferimento" datetime="data di inserimento"  
lang="lingua">Testo inserito</ins>
```

43) il Tag **<bdo>** serve a indicare la direzione del testo. Da sinistra a destra (ltr), da destra a sinistra (rtl).

```
<bdo dir="ltr | rtl" lang="lingua">Testo</bdo>
```

```
30af04ad110f1f5d
```

44) il Tag **<button>** permette di inserire nella pagina un pulsante con il contenuto indicato. Esistono tre tipi di pulsanti: normale (type="button"), reset (type="reset"), submit (type="submit").

`<button accesskey="tasto di scelta rapida" disabled="disabled" type="button | reset | submit">Testo o immagini</button>`

45) il Tag **<fieldset>** viene utilizzato all'interno di un form per raggruppare vari elementi tra loro.

`<fieldset lang="lingua" title="titolo informativo">Definizione dei campi</fieldset>`

46) il Tag **<form>** introduce un form interattivo. Per default enctype assume il valore application/x-www-form-urlencoded. L'attributo target assume come valori: _blank (apre una nuova pagina), _parent (indica il frame principale), _self (indica lo stesso frame) e _top. Action assume get per l'invio di dati codificati nell'URI e post per dati inviati all'interno della richiesta HTTP.

`<form lang="lingua" accept="set di caratteri di input" enctype="tipo di contenuto da inviare al server" action="URI a cui inviare i dati" method="get | post" target="frame di visualizzazione dei risultati"> Definizione del form</form>`

47) il Tag **<input/>** Introduce un controllo all'interno di un form di interazione con l'utente.

`<input type="text | password | checkbox | password | radio | image | reset | button | hidden | file" name="nome del controllo" value="valore" accept="MIME type accettati" checked="checked" disabled="disabled" readonly="readonly" size="dimensioni del controllo" maxlength="numero massimo di caratteri" />`

48) il Tag **<label>** ha la funzione di inserire il testo descrittivo di un controllo all'interno del form.

`<label for="id del relativo controllo" lang="lingua" title="testo informativo" />Etichetta</label>`

49) il Tag **<legend>** ha la funzione di inserire un titolo per un gruppo di controlli definiti attraverso il tag <fieldset>.

`<legend lang="lingua" title="titolo informativo">Titolo</legend>`

50) il Tag **<optgroup>** E' utilizzato all'interno di una lista di elementi **<option>** per raggruppare una serie di voci correlate logicamente.

```
<optgroup lang="lingua" label="descrizione" disabled="disabled" /gt;
```

51) il Tag **<option>** viene utilizzato per definire gli elementi presenti in una lista di tipo **select**.

```
<option label="testo descrittivo" lang="lingua"  
selected="selected">Elemento</option>
```

52) il Tag **<select>** introduce una lista di selezione (selection list).

L'attributo **disabled** permette di disattivare la possibilità di interazione con l'utente, l'attributo **multiple** consente all'utente di compiere scelte multiple, infine l'attributo **size** indica il numero di elementi che vengono visualizzati.

```
<select disabled="disabled" multiple="multiple" name="nome" size="dimensioni"  
title="testo informativo">Contenuto</select>
```

53) il Tag **<textarea>** è un campo di input a linee multiple che consente all'utente l'inserimento di testo.

```
<textarea cols="numero di colonne" rows="numero di righe" lang="lingua"  
disabled="disabled" readonly="readonly" name="nome" title="testo informativo">Testo di  
default</textarea>
```

54) il Tag **<caption>** definisce la didascalia di un immagine o di una tabella.

```
<caption lang="lingua">Testo della didascalia</caption>
```

55) il Tag **<col>** introduce ogni singolo elemento definito in una sezione **<colgroup>**, **span** indica il numero di colonne convenzionali su cui deve estendersi l'elemento **<col>**.

```
<col span="numero di colonne" />
```

56) Il Tag **<colgroup>** Questo tag viene utilizzato all'interno di una tabella per raggruppare un certo numero di colonne.

```
<colgroup char="carattere di allineamento" span="numero di colonne"
width="larghezza"> </colgroup>
```

57) il Tag **<table>** viene utilizzato per definire una tabella.

```
<table>Definizione della tabella</table>
```

58) il Tag **<tbody>** è utilizzato all'interno della definizione di una tabella per raggruppare un certo numero di celle e applicare una proprietà comune.

L'attributo char indica che l'allineamento deve essere relativo a un certo carattere (ad esempio char="," per i numeri decimali si ottiene l'allineamento rispetto alle virgole).

L'attributo charoff indica il numero di caratteri da aggiungere rispetto all'allineamento.

```
<tbody char="carattere di allineamento" charoff="distanza dal carattere di
allineamento"></tbody>
```

59) il Tag **<td>** definisce una colonna all'interno di una riga di una tabella.

```
<td colspan="numero di colonne occupate" rowspan="numero di righe occupate"
char="caratteri di allineamento" charoff="distanza dal carattere di
allineamento">Contenuto della cella</td>
```

60) il Tag **<tfoot>** compare nella definizione di una tabella e ha la funzione di definire una o più righe che costituiscono la parte inferiore di una tabella.

```
<tfoot char="carattere di allineamento" charoff="distanza dal carattere di
allineamento">Testo</tfoot>
```

61) il Tag **<th>** Ha una funzione e una sintassi analoga al tag td. La differenza consiste nel fatto che questo tag introduce gli elementi di intestazione di righe e colonne.

62) il Tag **<thead>** compare nella definizione di una tabella per raggruppare alcune linee che costituiscono la sezione superiore. Ha una sintassi analoga al tag tfoot.

63) il Tag **<tr>** ha la funzione di delimitare una riga nella definizione di una tabella. Ha una sintassi analoga al tag td.

64) il Tag **** ha la funzione di includere un immagine all'interno della pagina.

```

```

65) il Tag **<area>** viene utilizzato per definire un area sensibile nelle mappe d'immagine. shape può assumere i seguenti valori poly (poligono), rect (rettangolo) e circle (cerchio).

```
<area coord="coordinate" shape="forma" href="collegamento"> </area>
```

66) il Tag **<map>** definisce una mappa cliccabile gestita lato client.

```
<map name="nome" title="testo informativo">Definizione degli elementi</map>
```

67) il Tag **<ismap>** indica che all'immagine è associata una mappa cliccabile lato server.

```
ismap="ismap"
```

68) il Tag **<meta/>** è posto nell'intestazione della pagina, tra i tag head, e permette di fornire informazioni accessorie relative al documento.

```
<meta name="nome della variabile" content="valore della variabile"
lang="lingua" http-equiv="intestazione http" />
```

69) il Tag **<noscript>** indica una porzione di codice che deve essere visualizzata se il browser non supporta il linguaggio di scripting utilizzato.

```
<noscript> Testo alternativo </noscript>
```

70) il Tag **<script>** è usato per isolare uno script lato client dal resto del documento.

```
<script charset="set di caratteri" src="URI dello script">Codice dello
script</script>
```

71) il Tag **<style>** è inserito nell'intestazione di un documento per indicare i fogli di stile.

```
<style lang="lingua" media="tipo di media" type="MIME type">Definizione del foglio di stile</style>
```

72) il Tag **<link/>** viene impiegato per definire relazioni tra il documento corrente e altri documenti presenti in rete.

Deve comparire necessariamente nella sezione head.

```
<link charset=set di caratteri" lang="lingua" dir="direzione del testo" href="URI" rel="tipo di relazione con il documento collegato; alternate | stylesheet | start | next | prev | contents | index | glossary | copyright | chapter | section | subsection | appendix | help | bookmark" rev="tipo di relazione tra documento collegato e documento; alternate | stylesheet | start | next | prev | contents | index | glossary | copyright | chapter | section | subsection | appendix | help | bookmark" title="testo informativo" media="tipo di media" type="tipo di contenuto">
```

73) il Tag **<base/>** indica l'URL di partenza a cui si riferiscono le URL relative presenti nella pagina. Questo tag deve essere inserito all'interno del tag head

```
<base href="URL di partenza" />
```

74) il Tag **<rb>** (ruby base) permette di effettuare un markup alla base del testo.

```
<rb>Testo</rb>
```

75) il Tag **<rbc>** (ruby base container) serve per raggruppare i tag rb.

```
<rbc>
...
</rbc>
```

76) il Tag **<rp>** può essere usato per specificare i caratteri che denotano l'inizio e la fine di un testo ruby.

```
<rp>Carattere iniziale</rp>
<rt>Testo</rt>
<rp>Carattere finale</rp>
```

77) il Tag **<rt>** permette di effettuare un markup al testo dell'elemento ruby.

```
<rt>Testo</rt>
```

78) il Tag **<rtc>** (ruby text container) include i tag rt.

```
<rtc>
...
</rtc>
```

79) il Tag **<ruby>** serve come contenitore per il ruby markup.

```
<ruby>
...
</ruby>
```

80) il Tag **<!-- ... -->** racchiude un commento nel codice del documento.

```
<!-- Testo del commento -->
```

81) **accesskey** (attributo) consente di specificare un tasto di scelta rapida.

```
accesskey="tasto"
```

82) **class** (attributo) permette di aggiungere all'elemento la classe del foglio di stile specificata.

83) **id** (attributo) permette di aggiungere all'elemento lo stile identificativo specificato.

84) **MimeType** (attributo) identifica il formato dei dati. Ogni MIME type si compone di due parti: una descrittiva e una particolare per il formato usato per la codifica.

85) **tabindex** Questo attributo consente di definire l'ordine di selezione degli elementi della pagina tramite il pulsante Tab.

EVENTI

onblur

Descrizione: Si verifica quando l'elemento perde l'acquisizione dell'input da tastiera.

onchange

Descrizione: Si verifica nel momento in cui cambia lo stato dell'oggetto.

onclick

Descrizione: Si verifica quando l'utente fa click sull'oggetto.

ondblclick

Descrizione: Si verifica quando l'utente fa doppio clic sull'oggetto.

onfocus

Descrizione: Si verifica quando l'elemento acquisisce l'input da tastiera.

onkeydown

Descrizione: Si verifica quando l'utente preme un tasto e l'elemento acquisisce l'input da tastiera.

onkeypress

Descrizione: Si verifica quando l'utente preme un tasto e l'elemento acquisisce l'input da tastiera.

onkeyup

Descrizione: Si verifica quando un tasto viene rilasciato mentre l'elemento acquisisce l'input da tastiera.

onload

Descrizione: Si verifica quando l'oggetto viene caricato.

onmousedown

Descrizione: Si verifica quando l'utente preme un pulsante del mouse sopra l'oggetto.

onmousemove

Descrizione: Si verifica nel momento in cui il puntatore viene spostato all'interno dell'oggetto.

onmouseout

Descrizione: Si verifica nel momento in cui il puntatore del mouse esce dall'oggetto.

onmouseover

Descrizione: Si verifica nel momento in cui il puntatore del mouse viene spostato sopra l'oggetto.

onmouseup

Descrizione: Si verifica nel momento in cui l'utente rilascia un pulsante del mouse sopra l'oggetto.

onreset

Descrizione: Si verifica nel momento in cui l'utente preme un pulsante di tipo reset.

onselect

Descrizione: Si verifica quando l'utente seleziona l'oggetto.

onsubmit

Descrizione: Si verifica nel momento in cui l'utente preme il pulsante di tipo submit.

CARATTERI SPECIALI

** **

Descrizione: Spazio.

!

Descrizione: Punto esclamativo.

—

Descrizione: Trattino.

&

Descrizione: & (e commerciale).

¦

Descrizione: |

Ç

Descrizione: Ç.

¸

Descrizione: ç.

¢

Descrizione: Simbolo dei centesimi.

©

Descrizione: Simbolo del copyright.

¤

Descrizione: ¤.

é

Descrizione: é.

É

Descrizione: É.

è

Descrizione: è.

È

Descrizione: È.

ë

Descrizione: ë.

€

Descrizione: Simbolo dell'Euro.

>

Descrizione: Maggiore (>).

¡

Descrizione: ¡.

<

Descrizione: minore (<).

µ

Descrizione: Simbolo del prefisso micro- (unità di misura).

** **

Descrizione: Spazio (non breaking).

ñ

Descrizione: ñ.

"

Descrizione: Virgolette.

®

Descrizione: Simbolo del marchio registrato.

™

Descrizione: Simbolo TM.

¥

Descrizione: Simbolo dello Yen.

BIBLIOGRAFIA

- 1) www.rhadamanth.net
- 2) www.html.it
- 3) www.manuali.it/ . com