

HTML+XML= XHTML

Il ritorno al futuro del WEB

A cura di Barbara Lotti

## Che cos'è l'XHTML

- Nel gennaio del 2000 il W3C ha rilasciato, anziché una nuova versione, una riformulazione dell'HTML come applicazione XML, l'XHTML (nell'ottobre 2001 ne viene rilasciata una definizione più esaustiva, l'XHTML 1.0).
- L'XML (eXtensible Markup Language) è un metalinguaggio che consente la creazione di infiniti linguaggi di marcatura, è uno strumento molto versatile che permette la condivisione totale dei dati, intesi come pura informazione, a prescindere dal tipo di visualizzazione e di utilizzo.
- L'XHTML eredita la versatilità dell'XML, garantendo l'accesso al web anche a dispositivi diversi dal computer, come telefonini, PDA, WebTV.
- L'XHTML è compatibile con l'HTML 4.0 e quindi con i browser per la navigazione sul web.
- Le maggiori differenze con l'HTML sono di ordine sintattico, infatti l'XHTML eredita la rigidità sintattica dell'XML e l'obbligo della validazione.

## Regole di scrittura

1. Tutti i tag ed i loro attributi devono essere in minuscolo (case sensitive)
2. Obbligatorio il tag di chiusura; di conseguenza i tag vuoti come `<br>` ed `<img>` diventano `<br />``<img />`
3. I valori degli attributi devono essere specificati fra doppi apici o singoli apici.
4. Ogni attributo deve avere un valore
5. Si deve utilizzare l'attributo **id** al posto di **name** per identificare gli elementi di un documento

## Vantaggi dell'XHTML

- **Codice pulito e ben strutturato**: con l'XHTML (versione strict, XHTML1.1) tutta la formattazione del documento è affidata ai CSS.
- **Portabilità**: tutta l'evoluzione dei servizi mobili sarà fondata sull'integrazione tra XHTML e CSS.
- **Estensibilità**: ciò significa che è possibile inserire in un documento parti scritte in uno dei tanti linguaggi della famiglia XML.
- **Accessibilità**: una pagina valida, ben definita nella struttura, è meglio gestibile da browser alternativi come quelli vocali o testuali.

## Versioni dell'XHTML

Ciascuna delle seguenti versioni prevede regole sintattiche diverse perché tutte fanno riferimento a distinte DTD (definizione di tipo di documento):

**XHTML Strict:** non sono ammessi tag e attributi relativi all'aspetto grafico del documento (font, color, align..), controllato dai fogli di stile.

**XHTML Transitional:** ammette tag di formattazione; è la versione più simile all'HTML 4.01.

**XHTML Frameset:** prevede l'utilizzo di frame.

**XHTML 1.1:** questa versione (l'ultima rilasciata dal W3C) si basa sulla DTD Strict della versione 1.0 e rappresenta la prima formulazione pratica del concetto di modularità. Gli elementi fondamentali (i 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.

## Validazione

- Un documento XHTML deve essere convalidato rispetto ad una delle tre DTD.
- Un documento XHTML è **ben formato** quando rispetta le regole della sintassi XML (presenza di un elemento radice, dichiarazione doctype, corretto annidamento dei marcatori, chiusura obbligatoria dei tag vuoti)
- Un documento XHTML è **valido** se usa correttamente gli elementi e gli attributi dichiarati nella sua DTD.
- All'indirizzo <http://validator.w3.org> è possibile convalidare qualsiasi documento

## Intestazione

- In un documento XHTML l'elemento radice deve essere **<html>**.
- L'elemento radice **<html>** deve contenere la dichiarazione di un namespace XML (spazio dei nomi) tramite l'attributo `xmlns`. Il namespace usato deve essere **`http://www.w3.org/1999/xhtml`**
- In un documento XHTML l'elemento radice deve essere preceduto da un elemento **<!DOCTYPE>**. All'interno di questo elemento è necessario specificare la DTD di riferimento e il suo URI.

## Dichiarare la DTD

- **DTD Strict**  
`<!DOCTYPE html  
PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">`
- **DTD Transitional**  
`<!DOCTYPE html  
PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">`
- **DTD Frameset**  
`<!DOCTYPE html  
PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">`
- **DTD XHTML1.1**  
`<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"  
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">`

la parola chiave **PUBLIC** significa che la DTD è pubblica, creata dal W3C; l'url indica che la DTD è collocata sul sito del W3C. Il **DOCTYPE**, non ha alcun effetto sulla presentazione della pagina. Serve solo al validatore per stabilire le regole della convalida.

## Esempio

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"> doctype
<html xmlns="http://www.w3.org/1999/xhtml"> elemento radice
<head>
<title>La mia pagina XHTML</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
dichiaro la codifica dei caratteri

</head>
<body>
<h2>Il ritorno al futuro del web</h2>
<p>L'XHTML sostituisce l'HTML</p>
</body>
</html>
```

## Elementi ed attributi

### DTD Strict

Si tratta della DTD più rigida, orientata alla sola struttura del documento: tutto ciò che concerne la presentazione del testo è affidata ai fogli di stile, di conseguenza tutti gli elementi e gli attributi di formattazione sono eliminati.

#### Elementi non ammessi:

`<applet>`, `<basefont>`, `<center>`, `<dir>`, `<font>`, `<frame>`, `<frameset>`, `<iframe>`, `<isindex>`, `<menu>`, `<noframes>`, `<s>`, `<strike>`, `<u>`

Occorre fare attenzione anche ad **attributi** generalmente ammessi in HTML:

- sono esclusi tutti gli attributi del tag `<body>`
- 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`)

## Elementi ed attributi

- **DTD Transitional**

Basata sull'omologa DTD di HTML 4.0 supporta tutti gli elementi e gli attributi di presentazione di HTML 4.0, anche quelli 'deprecati' dal W3C. pertanto ammesso l'uso delle tabelle per il layout e del tag <font>.

- **DTD Frameset**

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

## XHTML1.1

XHTML 1.1 (rilasciato il 31 maggio 2001) è un linguaggio di markup orientato alla sola struttura del documento: si basa infatti sulla DTD Strict di XHTML 1.0 e tutti gli elementi e gli attributi di presentazione sono definitivamente esclusi.

La DTD XHTML 1.1 è diversa dalle precedenti. Non contiene una lista di elementi e attributi con le regole che ne definiscono l'uso. È invece costituita da diverse dichiarazioni che includono altrettanti moduli. Le specifiche sui moduli XHTML fanno parte di un'altra raccomandazione, quella sulla [modularizzazione di XHTML](#).

Le differenze con la DTD Strict di XHTML 1.0 sono:

- l'attributo lang è sostituito da xml:lang
- per gli elementi img e map l'attributo name è sostituito da id
- è stata inserita la collezione di oggetti ruby

Rimangono invariate le altre regole di base relative all'elemento radice <html> e alla dichiarazione del namespace con l'attributo xmlns.

## Il namespace

- L'elemento **<html>** può assumere questi attributi:
  - **Dir:** determina la direzione del testo
  - **Lang:** specifica il linguaggio di base dell'elemento quando è interpretato come HTML
  - **xml:lang** specifica il linguaggio di base dell'elemento quando è interpretato come XML
  - **XmInS:** specifica il namespace predefinito per XHTML
  - L'unico attributo con valore obbligatorio è **xmInS:** "http://www.w3.org/1999/xhtml".
- La necessità di un namespace (spazio di nomi) dipende dalla derivazione da XML, linguaggio estensibile. È possibile, infatti, estendere il set di tag di XHTML con elementi di altri linguaggi, anche creati personalmente.

## Testata

- La funzione principale della sezione **<head>** è quella di contenere informazioni che non vengono direttamente visualizzate nella pagina, ma che sono comunque di grande rilievo.
- Elementi che possono apparire nella testata:
  - **<base>** Usato per definire l'URL di base della pagina; utile per la risoluzione dei link relativi.
  - **<link>** Contiene informazioni su documenti esterni collegati. Usato soprattutto per i CSS.
  - **<meta>** Specifica informazioni di vario tipo sul documento.
  - **<noscript>** Usato per visualizzazioni alternative nei browser che non supportano gli script.
  - **<object>** Racchiude un oggetto.
  - **<script>** Contiene script di programmazione .
  - **<style>** Definisce le regole di formattazione per il documento corrente
  - **<title>** Specifica il titolo del documento che compare nella barra del titolo del browser

## Corpo del documento

Il documento è compreso (come in HTML) fra **<body>** di apertura e **</body>** di chiusura. Gli attributi per il testo (es. colore dei link, il colore di sfondo) sono vietati solo nella DTD Strict. Non vanno pertanto usati e devono essere sostituiti dai CSS i seguenti attributi:

- `alink`
- `background`
- `bgcolor`
- `link`
- `text`
- `vlink`

La correttezza formale in XHTML è obbligatoria: i tag vanno, oltre che ben annidati, inseriti correttamente nel testo, rispettando le gerarchie strutturali e formali. Ciò significa che fra gli elementi strutturali `<div>` deve precedere `<p>` (`<div><p> La correttezza formale in XHTML è obbligatoria </p></div>`) e non viceversa; che un marcatore di presentazione come `<b>` non può contenere al suo interno un tag di struttura come `<p>`, ma il contrario (`<p><i>La sintassi dell'XHTML</i></p>`)

## La correttezza formale

In XHTML non sono stati introdotti nuovi elementi o attributi rispetto ad HTML 4.01: la differenza principale sta nell'applicazione di regole sintattiche più rigide, derivate da XML. Se tali regole sintattiche non vengono rispettate (regole che i browser non segnalano come errore) non si otterrà un documento XHTML valido.

Le ricordiamo:

1. Corretto annidamento dei marcatori
2. I nomi di elementi e attributi devono essere in minuscolo
3. Ciascun attributo deve essere posto tra virgolette e avere indicato il valore.
4. Tutti gli elementi devono essere chiusi; per questo gli elementi vuoti devono terminare con `/>`
5. Per identificare un elemento si deve usare l'attributo `id` e non `name`.

## Elementi critici

In XHTML vengono ereditate da HTML alcune regole che proibiscono l'annidamento di determinati elementi.

- un elemento <a> non può contenere altri elementi <a>
- l'elemento <pre> non può contenere gli elementi <img>, <object>, <big>, <small>, <sub>, <sup>
- l'elemento <button> non può contenere <input>, <select>, <textarea>, <label>, <button>, <form>, <fieldset>, <iframe>, <isindex>
- l'elemento <label> non può contenere un altro elemento <label>
- l'elemento <form> non può contenere l'elemento <form>
  
- Se si fa uso della e commerciale (&) in valori di attributi è preferibile esprimerlo con un riferimento ad una entità di carattere: “&amp;” (è il caso di URL assolute copiate in funzione di link).
  
- Quando si collegano CCS esterni è opportuno che questi utilizzino le minuscole (i fogli di stile e gli script vanno inseriti fra <head> di apertura e </head > di chiusura).

## Riferimenti utili

- [XHTML Reference di W3Schools](#)
- [Specifica XHTML 1.0](#) tradotta in italiano
- [Specifica XHTML 1.1](#).
- [Introduzione alla modularizzazione](#): dal sito del W3C
- Il concetto di [namespace](#) in XML
- [Lista di elementi strutturali](#) o di blocco dal sito HTML.it
- [Lista di elementi di presentazione](#) o inline dal sito HTML.it