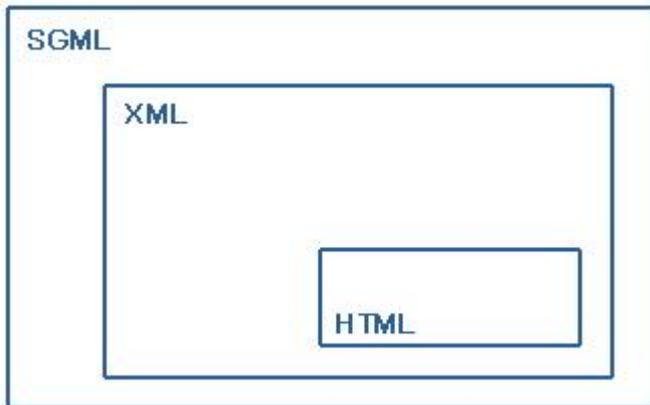


## Sviluppo di pagine e siti

### Creare pagine Web

**HTML** è l'acronimo di *HyperText Markup Language*; si tratta di un linguaggio utilizzato per la marcatura ipertestuale. A differenza dei comuni linguaggi di programmazione, **HTML** non serve a implementare programmi, funzioni o *subroutine* in grado di tradurre un algoritmo di calcolo, ma viene impiegato per definire le caratteristiche di pagine ipertestuali. Le istruzioni di **HTML** impostano gli attributi relativi agli oggetti utilizzati nella pagina, ovvero ne determinano la posizione, il formato, la visibilità e le altre proprietà che risulta possibile impostare.



**HTML** si basa sulla sintassi dello *Standard Generalized Markup Language (SGML)*, un metalinguaggio per la definizione di sistemi di marcatura. Attraverso il linguaggio **HTML** è possibile inserire gli iperlink, ovvero dei collegamenti ancorati a oggetti posizionati all'interno della pagina (testuali e non), che permettono di saltare da una pagina all'altra del medesimo sito o di siti remoti. Risulta allora evidente che **HTML** permette di realizzare un ipertesto, ovvero un testo la cui lettura tipicamente può procedere secondo diversi percorsi determinati dal programmatore, ma scelti di volta in volta dal fruitore delle pagine.

In questa parte saranno discussi concetti introduttivi di **HTML**. Per i concetti più avanzati si rimanda all'appendice.

### Usare HTML per creare e aggiungere pagine Web

Durante la scrittura di una pagina occorre seguire le regole sintattiche del linguaggio. Tuttavia, **HTML** è presente in **Internet** in diversi dialetti, ciascuno dei quali risulta ottimale per un determinato *browser*. In tal modo la stessa pagina potrà essere visualizzata in modo diverso da *MS Internet Explorer* o da *Netscape Navigator*. Può accadere inoltre che una determinata struttura sintattica (si parlerà di marcatori, o tag) non sia riconosciuta affatto da alcuni *browser*. Anche in questo caso la pagina sarà visualizzata senza alcuna generazione di errori, ma semplicemente ignorando l'elemento sconosciuto. Si usa dire a proposito del linguaggio **HTML**, che occorre essere rigorosi nella scrittura del codice letto da *browser* che risultano essere tolleranti.



Così come avviene per linguaggi di programmazione di alto livello, anche la scrittura di una pagina **HTML** può essere realizzata attraverso l'uso di indentazione o altre modalità che facilitino la lettura del codice. Tuttavia il *browser* non interpreterà invii a capo o inserimento di spazi multipli (a meno che si faccia uso di un particolare marcatore di preformattazione o si inseriscano 'fisicamente' degli spazi), leggendo il testo scritto sequenzialmente senza soluzione di continuità. Come detto se vengono inseriti più spazi, essi vengono visualizzati in un solo elemento. Si parla cioè di spazi collassati.

### Struttura della pagina

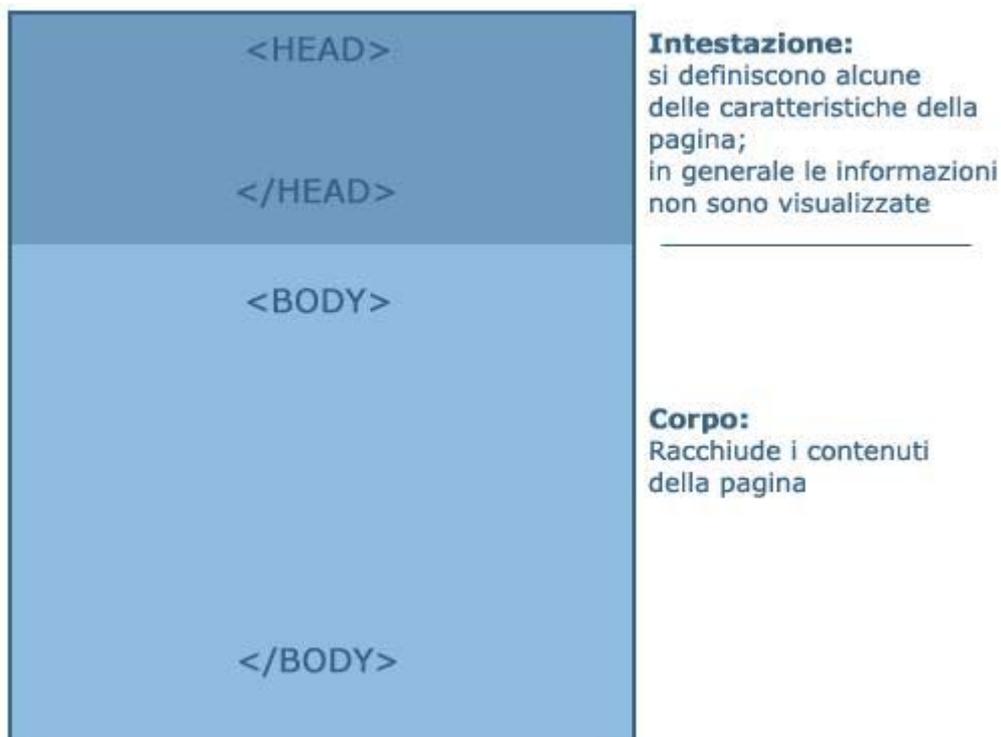
Una pagina ipertestuale scritta in **HTML** si può considerare composta da due sezioni: una parte di intestazione, all'interno della quale vengono definite alcune caratteristiche generali della pagina, e il corpo della stessa pagina, ove si troveranno tra l'altro i contenuti da visualizzare. Le due sezioni devono essere scritte in modo sequenziale e racchiuse da marcatori di inizio e fine sezione; entrambe le sezioni saranno quindi delimitate da marcatori di inizio e fine pagina. Un marcatore è inserito attraverso la propria sintassi, che prevede la digitazione del proprio nome tra parentesi angolari; per l'elemento di chiusura bisognerà far precedere il nome dal simbolo /.

È possibile inoltre inserire dei commenti, ovvero delle descrizioni che non hanno effetto sulla visualizzazione della pagina. Per inserire un commento occorrerà utilizzare la seguente sintassi.

```
<!-- stringa di commento -- >
```

In sostanza la struttura di una pagina **HTML** è la seguente.

```
<HTML>
```



Come per alcuni linguaggi di programmazione tradizionali, il codice **HTML** va scritto preferibilmente usando una indentazione che tuttavia non ha alcun effetto sulla presentazione della pagina, ma può risultare uno strumento che facilita la lettura del listato (per esempio aumentando il grado di indentazione quando si passa a un livello inferiore di marcatura, ovvero si inserisce un tag che deve essere incluso nel precedente).

I marcatori principali

- **Il marcatore di pagina.** I marcatori di inizio e fine pagina sono rispettivamente i seguenti:

```
<HTML> </HTML>
```

tale marcatore ha la funzione di delimitare gli elementi costitutivi della pagina stessa. Tutti gli altri marcatori (compresi quelli di intestazione e di corpo) dovranno essere compresi tra tali tag.

- **Il marcatore di intestazione.** La sezione di intestazione è delimitata dai marcatori:

```
<HEAD> </HEAD>
```

Si è detto che l'intestazione serve a definire alcune caratteristiche generali, che per la gran parte non risultano visibili all'interno della pagina o non risultano comunque indispensabili. Tra i marcatori di intestazione possono comunque essere inserite le seguenti informazioni: Titolo, Meta **Tag**, Elementi di stile, *Script*. Gli stili permettono di impostare le caratteristiche che riguardano il formato degli oggetti contenuti in più pagine. Gli *script* sono frammenti di codice che implementano funzioni diverse, come per esempio quelle che forniscono dinamicità alle pagine. All'interno dell'intestazione possono essere inseriti i marcatori che servono a porre informazioni non visibili all'interno del corpo della pagina; rientrano in questa categoria il titolo, i marcatori che definiscono caratteristiche di pagina (META per la definizione delle parole chiave, eccetera), elementi di stile o *script* che devono essere posti in esecuzione.

- **Il marcatore di titolo.** I marcatori di inizio e fine titolo sono rispettivamente i seguenti.

```
<TITLE> </TITLE>
```

Questo marcatore dovrà essere posto internamente al marcatore di intestazione. Con questo si definirà la stringa che sarà visualizzata nella barra del titolo del *browser*. Alcuni motori di ricerca utilizzano il contenuto del marcatore di titolo per ricevere informazioni (es. *hot words*) sulla pagina corrente.

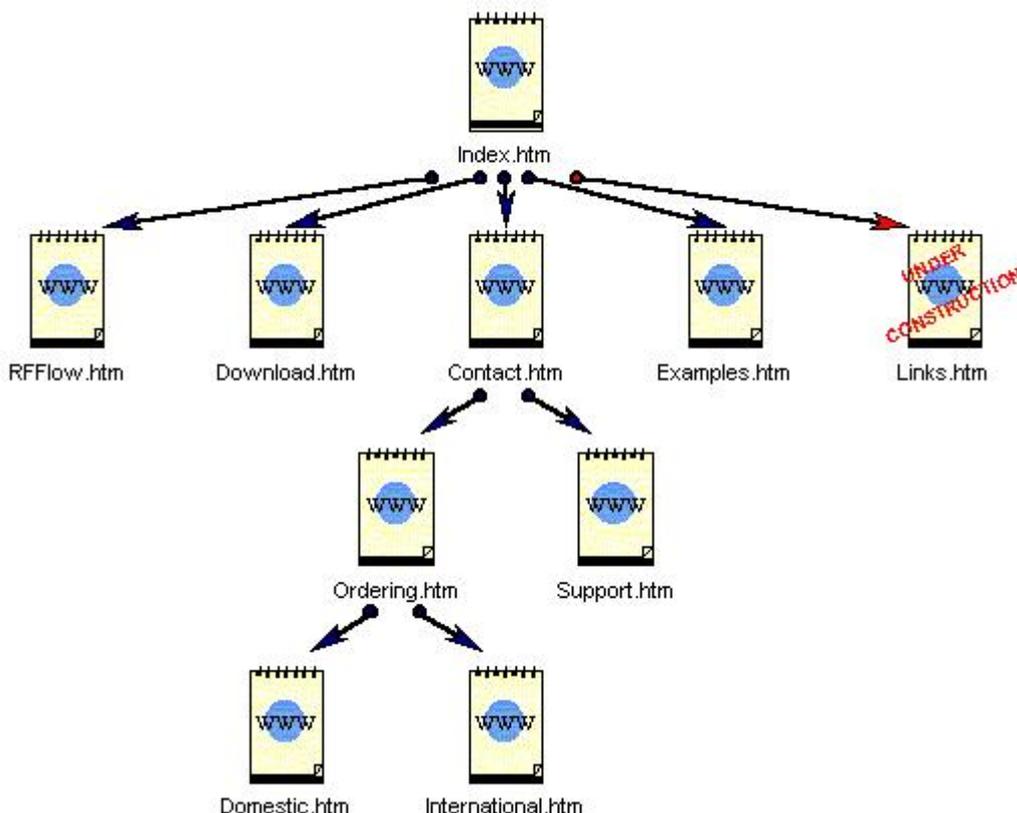
## Link

L'uso dei *link* permette di effettuare riferimenti a *file* remoti o ad altre parti del documento stesso (ad esempio per spostarsi velocemente da un indice alla voce interessata). Questo tag serve perciò a creare *link* tra il documento e un immagine o animazione o *file* sonoro. La sintassi del **Tag** è:

```
<A HREF ="file_name" (in formato URL)> testo che fa da collegamento </A>
```

Esempio: `<A href="prova.html"> Questo è il link a prova.html </A>`

Questo esempio crea un *link* con il *file* ipertestuale prova.html. Nel documento il *link* appare come la frase Questo è il *link* a prova.html sottolineata o evidenziata (a seconda del **Browser**) cliccando sulla frase verrà caricato il nuovo documento prova.html. Si consiglia di usare sempre *path* relativi in caso di spostamento dei *file* si avranno meno problemi.



## Formato dell'URL

Per effettuare il richiamo di qualsiasi *file* in Rete e non, si usa il formato **URL** (*Uniform Resource Locator*). Il formato è così composto:

Risorsa://SERVER[:PORT]/Path/Filename.

dove:

1. Risorsa é una delle seguenti:
  - **file** = specifica che l'oggetto in esame é un *file* e verrà prelevato dal *filesystem* locale del *browser*.
  - **http** = *file retrieval* da un *WWW server*.
  - **gopher** = *file retrieval* tramite un *Gopher server*.
  - **WAIS** = *file retrieval* tramite *WAIS server*.
  - **news** = *news server*.
  - **telnet** = *file retrieval* via *telnet*.
  - **mailto** = lancia l'applicazione per la posta elettronica con destinatario l'argomento della risorsa.
  - **ftp** = *ftp server*, consente di scaricare *file* residenti su siti remoti.
2. **SERVER** é l'indirizzo *IP* (o nome *DNS*) dell'*host* dove il *file* è localizzato.
3. **Port** è la porta da usare nella connessione. Opzionale.

Riassumendo: la prima parte dell'**URL** (prima delle due sbarre (//)) specifica il metodo di accesso. La seconda é tipicamente l'indirizzo dove sono locati il *computer*, i dati o i servizi richiesti. Le parti successive specificano gli eventuali nomi di *file*, di porte di connessione o il testo da cercare in un *database*. Ecco alcuni esempi di **URL**:

- file://www.iol.it/suono.au
- file://www.iol.it/picture.gif
- file://www.iol.it/directory
- http://www.iol.it/directory/book.html
- ftp://www.iol.it/pub/file.txt
- gopher://www.iol.it
- telnet://www.iol.it:1234
- news:alt.hypertext
- mailto:mariorossi@libero.it

La maggior parte dei *browser Web* concedono all'utente di specificare un **URL** e di connettersi a quel documento o servizio. Quando si seleziona dell'ipertesto in un documento **HTML** l'utente sta in effetti spedendo la richiesta di aprire un **URL**. In questo modo gli *hyperlink* possono essere fatti non solo ad altri testi e *media* ma anche ad altri servizi di rete. I *browser Web* non sono dei semplici *client* del *Web*, ma sono anche dei *client* totalmente operativi per i sistemi *FTP*, *Gopher* e *telnet*.

### Ancore e *Links* a parti specifiche di un documento

I *link* o le ancore possono essere anche utilizzate per spostarsi all'interno dello stesso documento facendo riferimenti a vari punti di questo. Supponiamo si voglia creare un *link* da un documento A ad una specifica sezione del documento B. Prima bisognerà marcare la sezione a cui poi si fara' riferimento nel modo seguente:

```
<A NAME="qui_Kirk">il testo che verra' visualizzato</A>
```

Quindi si creerà il *link* cosi':

```
<A HREF="...filename#qui_Kirk">...testo_visualizzato...</A>
```

Ove ...filename é l'**URL** che localizza il documento B e ..testo\_visualizzato.. é il testo (mostrato nel

documento A) che farà da *link*.