

1. Creare Il Tuo Primo Documento HTML

Obiettivi

Dopo questa lezione sarai in grado di:

- Identificare il significato e lo scopo dei tag HTML.
- Aprire uno spazio di lavoro per creare nuovi documenti HTML.
- Usare un editor di testo per creare le strutture HTML di base per ogni web page.
- Inserire commenti non visualizzati nei tuoi file HTML.
- Aprire i tuoi documenti nel tuo web browser per vedere come questi vengono visualizzati.

Lezione

Ora che sai cos'è l'HTML, cominciamo ad usarlo, no?

(Piccolo quiz -- cosa significa questa sigla?).

Cosa sono i tag HTML?

Quando un web browser visualizza una pagina come quella che stai leggendo ora, la legge da un file di solo testo, e va alla ricerca di codici speciali o "tags" che sono contrassegnati dai segni < e >. Il formato generale per un tag HTML è':

```
<nome_tag>stringa di testo</nome_tag>
```

Per esempio, il titolo per questa lezione usa un tag header:

```
<h3>Cosa sono i tag HTML?</h3>
```

Questo tag comunica al web browser di visualizzare il testo **Cosa sono i tag HTML?** nello stile di intestazione di livello 3 (su queste cose ne impareremo di più' andando avanti). I tag HTML possono dire al web browser di rendere il testo in grassetto, in corsivo, trasformarlo in un'intestazione, o farlo diventare un link ipertestuale ad un'altra pagina web. È importante notare che il tag di chiusura,

```
</nome_tag>
```

contiene il carattere "/" slash. Questa "/" barretta indica al web browser dove deve terminare il testo contrassegnato dal tag aperto in precedenza. Molti tag HTML sono accoppiati in questo modo. Se ti dimentichi la barretta, il web browser continuerà a considerare il tag valido per tutta la restante parte del tuo documento, dando risultati indesiderati (come in un esperimento che potresti provare a realizzare più tardi).

NOTA: Un web browser non fa caso al fatto che tu possa usare maiuscole o minuscole. Per esempio,

```
<h3> . . . </h3> non è reso diverso da <H3> . . . </H3>
```

Diversamente dalla programmazione informatica, se fai un errore tipografico in HTML, non ci sarà una distruzione del programma od un crash nel sistema; semplicemente, la tua pagina web sembrerà... sbagliata. È facile e veloce ritornare al codice HTML ed apportare le giuste modifiche.

Il tuo browser ha un vocabolario piccolo ma aperto! Un aspetto interessante dell'HTML è' che se il browser non riesce a capire come si deve comportare con il tag dato, lo ignora! Per esempio, nel documento che stai osservando, il tag d'intestazione di questa sezione assomiglia *veramente* a questa:

```
<wigggle><h3>What are HTML tags?</h3></wigggle>
```

ma dato che il tuo programma probabilmente non supporta questo <wigggle> tag (noi l'abbiamo creato, forse in futuro farà' in modo che il testo ondeggi lungo lo schermo?), esso procede con le sue conoscenze. Se stessi programmando un nuovo web browser, potresti decidere di aggiungere le funzionalità per il tag <wigggle> nel mio software.

Aprire Il Tuo Workspace

Per completare le lezioni di questo corso, dovresti creare una seconda finestra web (questo ti permette di tenere a portata di mano questa finestra con le istruzioni e la lezione ed una finestra come "spazio di lavoro"), più' aprire un'applicazione del tuo editor di testo in una terza finestra.

NOTA: Questo è' un buon punto per ricordarti che ti forniremo direttive che sono comunque generali come il

nome dei menu e dei file, che potrebbero variare in base al browser che stai usando. Se le nostre istruzioni dicono "Seleziona Apri... dal Menu File" ed il tuo web browser non ha questa *esatta* possibilita', cerca di trovare l'equivalente piu' simile.

Cosi' potrai essere in grado di saltare confortevolmente da una finestra ad un'altra applicazione diversa sul tuo computer. Un'altra opzione e' quella di stampare le istruzioni della lezione (ma noi non abbiamo intenzione di promuovere troppo quest'esagerato sfruttamento delle piante).

Ecco qui i passaggi necessari alla creazione di un "workspace":

1. Dal menu **File** del tuo web browser, seleziona **Nuovo/Finestra** o **Nuovo Web Browser** (Il nome esatto del comando da menu potrebbe essere diverso in base a quale tipo di programma di stia usando). Una seconda web window dovrebbe apparire. Pensa alla prima finestra come se fosse il tuo "libro di testo" ed al secondo clone come al tuo "workspace" per completare le lezioni di HTML.

NOTA: L'unica ragione per lavorare con due finestre e' che cosi' facendo puoi leggere le istruzioni per le lezioni e vedere al contempo il documento che stai creando. Non e' obbligatorio avere le due finestre aperte; rende solo il lavoro piu' semplice. Puoi anche mettere un segnalibro a questa pagina web o saltare indietro fino a qui attraverso il menu Vai a... o Cronologia.

2. Successivamente, avrai bisogno di saltare fuori dal tuo web browser ed aprire il tuo programma di text editor.

NOTA: Avrai necessita' di saltare indietro ed avanti tra finestre diverse per completare queste lezioni. Questa potrebbe essere una difficolta', in base alle dimensioni del tuo monitor. Potresti scegliere di ridimensionare le tre finestre in modo che si adattino al tuo schermo o farne una pila in modo che all'occorrenza tu possa portarle in primo piano semplicemente con un click.

Se stai usando un programma di word processor per creare il tuo codice HTML, assicurati di salvarlo in formato solo testo (od ASCII).

Se sei proprio un principiante ti raccomandiamo CALDAMENTE di usare l'editor di testo piu' semplice disponibile-- *SimpleText* o *TeachText* per gli utenti Macintosh od il *NotePad* di Windows. Perche' non usare questi comodi editor HTML? Sembra un approccio istruttivo che tu prima debba imparare i concetti e POI cercare le scorciatoie o gli aiuti che rendono piu' facile la vita ed il lavoro meno tedioso.

Ancora, ti sara' d'aiuto la creazione di una nuova directory/folder sul computer da usare come area di lavoro. La potresti chiamare *workarea* o *myspace* o in qualsiasi altro modo; solo assicurati di conservare tutti tuoi file creati in questo unico spazio. Ti semplifichera' la vita... almeno mentre lavori a questo tutorial!

Creare I Tuoi Documenti HTML

Un documento HTML contiene due parti distinte, il corpo e la testa. L'**Head** e' la parte che contiene informazioni riguardo il documento che non sono visualizzate sullo schermo. Il **body**, invece, e' costituito da tutto quello che e' visualizzato come parte della pagina.

La struttura di base allora di ogni pagina HTML e':

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
<html>
<head>
<!-- informazioni d'intestazione usate per fornire
      notizie extra riguardo questo documento,
      non visualizzate sulla pagina
-->
</head>

<body>

<!-- tutto l'HTML da visualizzare -->
      :
      :
      :
</body>
</html>
```

La primissima linea:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
```

Non e' tecnicamente richiesta, ma e' un codice che informa il browser per quale versione di HTML e' stata scritta la pagina corrente. Racchiudi tutto il lavoro in HTML nei tag `<html>...</html>`. Dentro, per primo viene il tag `<head>...</head>` e poi `<body>...</body>`. La tua web page puo' apparire gradevole sulla maggior parte dei computer senza nessuno di questi tag. In ogni caso, usandoli, le tue web page saranno in perfetto accordo con gli standard HTML Internazionali ed assicurati per la compatibilita' con i browser del prossimo futuro.

E' una buona abitudine, come quella di lavarti i denti :o).

Nota anche i tag di **commento** racchiusi da `<!-- blah blah blah -->`. Il testo contenuto tra questi tag NON viene visualizzato nella tua pagina web ma e' per informazioni che potrebbero essere utili a te o a qualsiasi altra persona che voglia

guardare a fondo il tuo codice sorgente HTML. Quando le tue pagine diventano complesse (come ti renderai conto quando ci avventureremo tra tabelle, frame, ed altre cosuccie divertenti, tra non meno di 20 lezioni!), i commenti diventeranno utilissimi nel momento in cui avrai bisogno di migliorare pagine create un sacco di tempo fa.

Questi sono i passi da compiere per creare il tuo primo file HTML. Sei pronto?

1. Se non l'hai ancora fatto, lancia il tuo programma di text editor.
2. Entra nella finestra del text editor.
3. Digita il seguente testo (NON devi schiacciare INVIO alla fine di ciascuna linea, ci pensera' il browser a dispiegare il testo e ad adattarlo):

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
<html>
<head>
<title>La mia prima pagina</title>
</head>
<!-- prova di pagina -->
<body>
In questa lezione userai la Rete per cercare
informazioni e poi scrivere il tuo
resoconto sui risultati.
</body>
</html>
```

NOTA: Guarda dove e' posizionato il tag <title>...</title>. E' nella parte contrassegnata da <head>...</head> e quindi non e' visibile sullo schermo. A cosa serve allora? Il tag <title> e' usato unicamente per identificare ogni documento ed e' anche visualizzato sulla barra degli strumenti del browser.

Nota anche che abbiamo inserito un tag di commento. Puoi scriverci quello che piu' ti pare tra i tags di commento, ma il tutto sara' visibile solo se si guarda nel sorgente HTML della pagina web.

4. Salva il documento come "prima.html" e conservalo nella folder/directory che fa da area di lavoro e che hai definito per questo corso. In piu', se stai usando un programma di word processor per creare il tuo codice HTML, assicurati di salvare come formato solo testo (o ASCII).

NOTA: Per gli utenti di Windows 3.1, si deve salvare tutti i codici HTML con nomi che terminino con l'estensione .HTM, ed in questo caso il tuo file sara' prima .HTM. Non preoccuparti!! Il tuo web browser e' abbastanza furbo da capire che i file con estensione .HTM sono file HTML.

Usando queste estensioni, un web browser sapra' come leggere questi file di testo come HTML e visualizzare adeguatamente la web page.

Visualizzare I Tuoi Documenti In Un Web Browser

1. Ritorna alla finestra web che stai usando per il tuo spazio di lavoro. (Se non hai ancora aperto una seconda finestra nel tuo browser, seleziona **Nuova Finestra** o **Nuovo Browser** dal menu **File**.)
2. Seleziona **Apri...** da **File**.
3. Usa la finestra di dialogo per trovare ed aprire il file che hai creato, "prima.html"
4. Ora, dovresti vedere nella barra del titolo della finestra dello spazio di lavoro il testo "Volcano Web" e nella pagina sotto, la sola frase <body> che hai scritto, "In questa lezione..."

2. Modificare un Documento HTML

Ora che hai creato il tuo primo documento HTML, imparerai come apportare modifiche ai tuoi documenti velocissimamente e a vederne i risultati nel tuo web browser.

Obiettivi

Dopo questa lezione sarai in grado di:

- Riaprire lo spazio di lavoro per la tua web page.
- Apportare cambiamenti al tuo documento HTML usando un text editor.
- Ricaricare il tuo documento nel web browser per apprezzarne i cambiamenti.

Lezione

Riaprire Il Tuo Spazio Di Lavoro

Per completare questa lezione, hai bisogno di creare una seconda finestra nel tuo web browser e riaprire la finestra di text editor che hai usato nella prima lezione. Seguono i punti da eseguire per riaprire il tuo workspace (ricorda che il nome esatto dei comandi dei menu potrebbero variare in base al browser che stai usando):

1. Crea una nuova finestra nel tuo web browser selezionando **Nuovo/Finestra** dal menu **File**.
2. Usa il comando **Apri...** dal menu **File** per trovare ed aprire il file HTML che hai creato nella lezione precedente.
3. Riapri il tuo programma di text editor.
4. Nel text editor, apri il file ("`prima.html`") che hai creato nella lezione precedente.

NOTA: Se sei un utente Windows 3.1, il tuo file dovrebbe chiamarsi "`prima.HTM`".

Da questo momento in poi, noi assumiamo tu sappia facilmente riaprire il tuo workspace in questo modo.

Apportare Cambiamenti Nel Tuo Documento HTML

1. Vai alla finestra del text editor.
2. Sotto il testo che hai digitato nella lezione precedente, schiaccia INVIO alcune volte e inserisci un testo.

Nota che questo testo dovrebbe essere **sopra** i tag `</body>` e `</html>` che sono in fondo al tuo file HTML.

3. Seleziona **Salva** dal menu **File** per rendere effettive le modifiche apportate al tuo file.

Ricaricare Il Tuo Documento Nel Web Browser

Ritorna al tuo workspace nel web browser, dove la versione precedente del tuo file era visualizzata. Nota che il nuovo testo che hai aggiunto non e' visibile. Per vedere i cambiamenti, usa il bottone **Aggiorna** o l'opzione nel menu specifica del tuo web browser. Questo obbliga il programma a rilegge lo stesso file e a visualizzare qualsiasi cambiamento sia avvenuto. Ora puoi vedere il nuovo testo che hai inserito.

Nota che il web browser ignora tutte le linee vuote e gli spazi extra (i ritorni a capo) che hai digitato nel file. In ogni caso, quando scrivi codice HTML, e' di grande aiuto separare le principali sezioni con linee vuote.... quando hai bisogno di tornare indietro ed inserire testo, ti rende piu' facile la localizzazione della posizione corretta in cui apportare le modifiche.

Naturalmente, ci saranno occasioni in cui **vorrai** che la tua pagina web abbia spazi vuoti tra le sezioni (per esempio tra paragrafi). Nella quarta Lezione impareremo come farlo.

Drag and Drop Bonus!

Ci *puo'* essere un modo piu' semplice per caricare e vedere pagine HTML. Dovrai sistemare il desktop del tuo computer in modo che tu possa vedere del tuo file HTML vicina alla finestra del tuo web browser. Devi semplicemente fare click e trasportare l'icona del tuo file "`prima.html`" o "`prima.htm`" dentro alla finestra del tuo web browser. Voilà! la tua pagina sara' visualizzata se il tuo computer supporta il drag and drop (funziona con i Macintosh OS 7.5 e Windows 9x).

Verifica

Argomenti di verifica per questa lezione:

1. Come hai riaperto il tuo workspace?
2. Quali passi hai fatto per apportare modifiche al tuo documento HTML?
3. Come hai visualizzato e visto questi cambiamenti nel tuo web browser?

Pratica Indipendente

Come hai fatto in questa lezione, modifica il tuo documento HTML personale che hai cominciato la scorsa lezione. Aggiungi poche frasi e vedi se riesci a ricaricare il documento modificato nel tuo web browser.

Prossimamente....

Ora che hai un'infarinatura di quelle che sono le tecniche di editing, aggiungeremo una sezione di intestazione grande e bella al tuo documento HTML.

3. Intestazioni: Sei Livelli Di approfondimento

Come puoi vedere in questa pagina, l'intestazione della sezione di questa pagina ("Intestazioni: Sei Livelli Di Approfondimento", "Obiettivi", "Lezioni", "Intestazioni HTML" ...) appare in dimensioni diverse e, forse, con colori e caratteri diversi. L'HTML fornisce tag per realizzare intestazioni a sei livelli decrescenti di significato. Il tuo browser determina l'esatto carattere tipografico e la grandezza giusta per la visualizzazione.

Obiettivi

Dopo questa lezione sarai in grado di:

- Identificare i diversi livelli di intestazione in HTML ed i tag a loro associati.
- Posizionare le intestazioni di vari livelli nei documenti HTML e vedere i cambiamenti col web browser.

Lezione

Intestazioni HTML

Le intestazioni sono realizzate in HTML "marcando" certi frammenti di testo con tag d'intestazione. Il formato per un tag d'intestazione HTML e':

```
<hN>Testo da Far Apparire nell'Intestazione</hN>
```

dove N e' un numero tra 1 e 6 per identificare le dimensioni dell'intestazione. Ecco alcuni esempi di diverse dimensioni d'intestazione:



I livelli d'intestazione variano da un livello di 1 (massima importanza) al livello 6 (minima importanza). Come principio, i livelli delle tue intestazioni dovrebbero avere un ordine logico costante.

Posizionare Intestazioni Nei Tuoi Documenti HTML

1. Riapri il tuo workspace (se non e' gia' aperto).
2. Vai alla finestra del text editor.
3. Apri il file HTML che hai creato nella lezione 2, "Volc.html".
4. Prima useremo il tag per visualizzare il titolo come intestazione massima, <H1>. Digita cio' che segue sopra il testo gia' esistente e **dopo** i tag </head><body>:

```
<h1>prima</h1>
```

5. Sotto il testo che hai gia' digitato, crea altre intestazioni per la futura sezione della tua pagina *Volcano Web*.

Digita le seguenti intestazioni nel body della tua web page (Nota che alcuni tag sono H3 ed altri sono H2):

```
<h2>Introduzione</h2>
```

```
<h2>Glossario</h2>
```

```

<h2>Luoghi Vulcanici negli USA</h2>
<h3>Mount St Helens</h3>
<h3>Long Valley</h3>
<h2>Luoghi Vulcanici su Marte</h2>
<h2>Progetti di Ricerca</h2>
<h3>Riferimenti</h3>

```

6. Salva le modifiche nel tuo text editor.
7. Ritorna al tuo web browser, **Apri** ed **Aggiorna** il file HTML.

Nota che sul computer che stai usando ora, puoi utilizzare i settaggi del tuo web browser per definire i caratteri e/o le dimensioni delle intestazioni. Per esempio su un computer, puoi avere che un browser visualizza i tag h1 con font Times a 36 punti; tag h2 come font Helvetica a 24 punti, etc. I codici HTML definiscono solo che le intestazioni debbano essere di un certo tipo (h1 a h6); come viene effettivamente visualizzato e' controllato dall'utente del web browser.

Controlla il tuo lavoro

Se alcune delle tue intestazioni sembrano non essere corrette, assicurati di controllare che i tag di chiusura e di apertura siano dello stesso livello.

Come esercizio opzionale, dai un'occhiata a quello che succede quando commetti un errore tipografico. Apri il tuo documento HTML nell'editor di testo e cancella lo slash (/) nel tag <h1>, dopo l'intestazione **Prima**:

```
<h1>Prima<h1>
```

Salva i cambiamenti Senza titolo e ricarica nel tuo web browser. Senza la conclusione corretta

Verifica

1. Quali sono i vari livelli d'intestazione in HTML?
2. Quali sono i tag associati a questi vari livelli?
3. Quali passaggi devi seguire per posizionare intestazioni nel tuo documento HTML?
4. Cosa succede se ti dimentichi uno slash alla fine di un tag d'intestazione?

Colori

I colori dello schermo sono definiti dai colori Rosso, Verde e Blu.

Perciò i colori in HTML sono definiti in un sistema basato sulla quantità di ognuno dei tre colori base.

Questo sistema si chiama RGB color system. (Ebbene sì - RGB significa Red, Green and Blue - Rosso, Verde e Blu!)

La maggior parte dei programmi di grafica (Photoshop, Photopaint, Paintshop Pro etc.) è in grado di lavorare con colori in un sistema RGB.

Il problema è che mentre gran parte dei programmi di grafica definiscono il codice RGB usando normali numeri decimali, HTML adopera i numeri esadecimali.

Questo significa che, se vuoi usare gli stessi colori nel tuo programma di grafica e sulla tua home page, dovrai convertire i numeri decimali in esadecimali.

Ecco una tipica definizione di colore in HTML:

```
<body bgColor="#FF8C67">
```

Il colore è definito come "#FF8C67".

Il simbolo # indica semplicemente che i seguenti numeri sono esadecimali.

I primi due caratteri specificano la quantità di rosso (FF).

I caratteri 3 e 4 specificano la quantità di verde (8C).

I caratteri 5 e 6 specificano la quantità di blu (67).

Qui sotto puoi vedere un esempio che mostra come i diversi colori vengono mescolati per ottenere il risultato desiderato:

FF + 8C + 67 = FF8C67

Essendovi due cifre esadecimali per ogni colore, si possono specificare 256 gradazioni di ogni colore base. Questo dà un totale di 256*256*256 = 16.777.216 differenti combinazioni.

Comunque, non tutti questi colori si vedranno allo stesso modo su tutti i browser e i sistemi operativi.

Soltanto 216 colori sono quelli che vengono considerati "sicuri" ("browser safe").

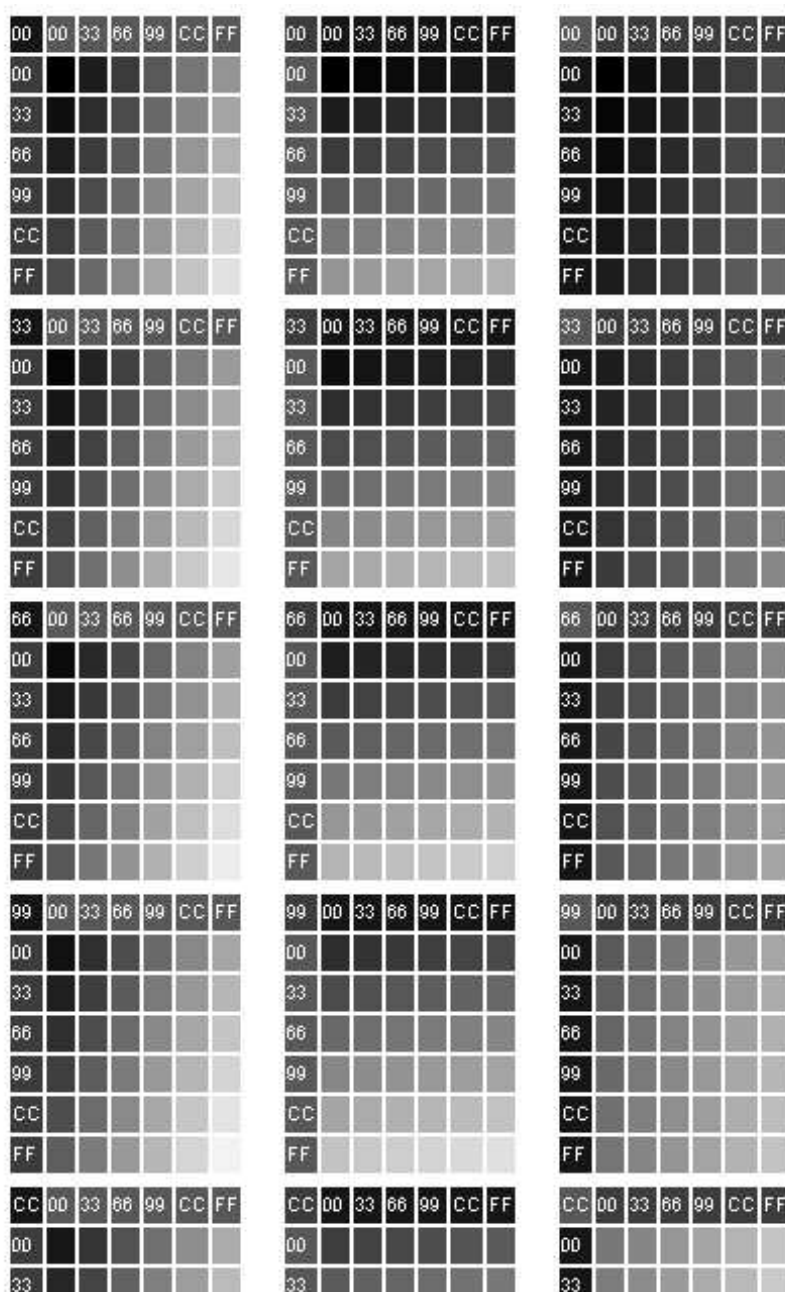
I colori sicuri sono colori che si vedono nello stesso modo, indipendentemente dal browser in cui appaiono.

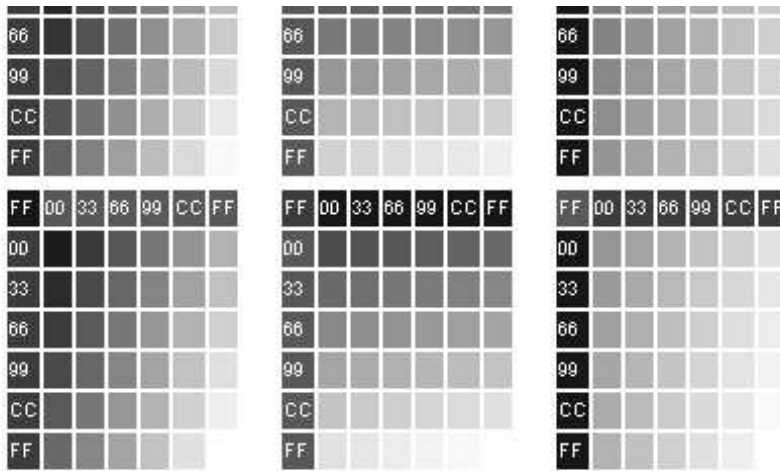
Il motivo è che i browser che girano su diverse piattaforme interpretano i colori diversamente.

Gli unici 216 colori che si vedono sempre e comunque allo stesso modo, sono quelli composti da coppie di 00, 33, 66, 99, CC e FF.

Un esempio potrebbe essere: "#990066"

Cerca di limitarti sempre ad usare i colori caratterizzati da queste combinazioni. Puoi usare la tavolozza per scegliere i colori sicuri.





Carattere, grandezza e colore del testo

Il tag FONT è uno dei più usati e frequenti nell'attuale Web publishing. Nella sezione: Impostare lo sfondo del documento (elemento BODY) abbiamo visto come l'attributo TEXT dia un colore univoco a tutto il testo del documento. Il tag FONT ha in parte una funzione simile, ma molto più ampia e concettualmente differente.

Il tag FONT ha la funzione di formattare tipo e grandezza del testo limitatamente ad alcuni punti del documento. In realtà se il tag font aprisse e chiudesse il documento, la totalità del testo compreso al suo interno sarebbe formattato di conseguenza. Ma concettualmente il tag FONT è stato pensato per definire parti limitate di testo.

Segue la sintassi corretta del tag:

```
<FONT FACE="arial" SIZE="5" COLOR="#000099">Carattere da formattare</FONT>
```

L'attributo **FACE="arial"** indica il tipo di carattere da richiamare per la visualizzazione.

SIZE="5" la grandezza del carattere che può andare da 1 a 7 (con maggiore grandezza dei numeri vicini al 7);

COLOR="#000099" indica il colore del testo (esprimibile anche in valori esadecimali).

, <I>

, <I> sono tre dei più usati tag di formattazione fisica dell'HTML. Tutti vanno aperti e successivamente chiusi:

 Testo grassetto

Il testo compreso tra questi tag è trasformato in grassetto (la B sta per BOLD).

<I> Testo corsivo </I>

Il testo compreso tra questi tag è formattato in corsivo (la I sta per ITALIC)

LISTE

Per creare una lista contrassegnata da punti devi inserire un tag e un tag all'inizio e alla fine della lista.

Le liste contrassegnate da numeri hanno i tag invece di .

Per separare le singole voci della lista, usa i tag e .

Ci sono particolari attributi che puoi adoperare per personalizzare le liste sulla tua pagina.

Nelle due pagine seguenti troverai notizie più dettagliate sulle liste a punti e su quelle numerate.

Questa pagina mostra come ottenere diversi tipi di liste a punti.

Puoi scegliere tra questi tipi di punti:

- disc
- circle
- square

Guarda questi esempi e il modo in cui sui ottengono.

| CODICE HTML | SPIEGAZIONE / ESEMPIO |
|---|--|
| <pre> text text text </pre> | <p>Crea una lista a punti usando il tipo di punto del default:</p> <ul style="list-style-type: none"> • text • text • text |
| <pre><ul type="disc"></pre> | <p>Inizia una lista a punti usando dischi come punti:</p> <ul style="list-style-type: none"> • Questa è una riga • Questa è un'altra riga • E questa è l'ultima riga |
| <pre><ul type="circle"></pre> | <p>Inizia una lista a punti usando cerchi come punti:</p> <ul style="list-style-type: none"> ○ Questa è una riga ○ Questa è un'altra riga ○ E questa è l'ultima riga |
| <pre><ul type="square"></pre> | <p>Inizia una lista a punti usando quadrati come punti:</p> <ul style="list-style-type: none"> ▪ Questa è una riga ▪ Questa è un'altra riga ▪ E questa è l'ultima riga |

Questa pagina mostra come si possono creare diversi tipi i liste a numeri.

Queste sono le opzioni numeriche disponibili:

- Numeri
- Lettere maiuscole
- Lettere minuscole
- Numeri romani maiuscoli
- Numeri romani minuscoli

In aggiunta a queste opzioni puoi specificare il numero di partenza della lista. Il valore di default per le liste a numeri è 1 (o la lettera A).

Guarda questi esempi e il modo in cui sui ottengono.

| CODICE HTML | SPIEGAZIONE / ESEMPIO |
|---|--|
| <pre> text text text </pre> | <p>Crea una lista a numeri usando il valore di default:</p> <ol style="list-style-type: none"> 1. text 2. text 3. text |

| | |
|--|---|
| <code><ol start="5"></code> | <p>Inizia una lista a numeri, partendo dal n. 5.</p> <ol style="list-style-type: none"> 5. Questa è una riga 6. Questa è un'altra riga 7. E questa è l'ultima riga |
| <code><ol type="A"></code> | <p>Inizia una lista a numeri, usando lettere maiuscole.</p> <ol style="list-style-type: none"> A. Questa è una riga B. Questa è un'altra riga C. E questa è l'ultima riga |
| <code><ol type="a"></code> | <p>Inizia una lista a numeri, usando lettere minuscole.</p> <ol style="list-style-type: none"> a. Questa è una riga b. Questa è un'altra riga c. E questa è l'ultima riga |
| <code><ol type="I"></code> | <p>Inizia una lista a numeri, usando numeri romani maiuscoli.</p> <ol style="list-style-type: none"> I. Questa è una riga II. Questa è un'altra riga III. E questa è l'ultima riga |
| <code><ol type="i"></code> | <p>Inizia una lista a numeri, usando numeri romani minuscoli.</p> <ol style="list-style-type: none"> i. Questa è una riga ii. Questa è un'altra riga iii. E questa è l'ultima riga |
| <code><ol type="1"></code> | <p>Inizia una lista a numeri, usando numeri normali.</p> <ol style="list-style-type: none"> 1. Questa è una riga 2. Questa è un'altra riga 3. E questa è l'ultima riga |
| <code><ol type="I" start="7"></code> | <p>Un esempio di come type (tipo) e start (inizio) possono essere combinati.</p> <ol style="list-style-type: none"> VII. Questa è una riga VIII. Questa è un'altra riga IX. E questa è l'ultima riga |

Esempio di lista + lista "nestata":

| | |
|---|---|
| <pre> ecco ecco testo 1 testo 2 testo 3 <ol type="a"> bla bla ecco </pre> | <pre> 1. ecco 2. ecco • testo 1 • testo 2 • testo 3 a. bla b. bla 3. ecco </pre> |
|---|---|

LINK

I link sono la parte più importante del world wide web. Sono i link che tengono insieme la rete.

Ci sono tre tipi di link che puoi avere nel tuo sito:

- Link che si agganciano alla pagina corrente (Interni).
- Link per altre pagine dello stesso sito (Locali).
- Link per pagine al di fuori del sito (Globali).

E' possibile creare testi e immagini funzionanti come link.

Questa sezione tratterà i link più "tradizionali": Testi e Immagini.

Altri tipi di link più avanzati richiedono l'uso di javascript.

I tag usati per generare i link sono `<a>` e `` .

Il tag `<a>` indica dove il link ha inizio, mentre `` ne indica la fine.

Tutto quello che è contenuto tra questi due tag si comporterà come un link.

La meta del link viene specificata aggiungendo al tag `<a>` usando l'attributo `href=" http://www.ecn.org "` .

L'esempio qui sotto ti mostra come far diventare la parola **qui** un link per ecn.org.

Clicca `qui` per andare su ecn.

Devi soltanto:

- Specificare la meta nello spazio `` .
- Inserire il testo che fa partire il link.
- Aggiungere infine un tag `` per indicare la fine del link.

Se devi fare un link a una pagina che fa parte del tuo sito e che quindi risiede sul tuo computer e poi sarà spostata sul server devi

specificare il percorso per raggiungere il file.

Clicca `qui` per andare sul file prova che è nella stessa cartella dove risiede il file a cui appartiene il codice .

Clicca `qui` per andare su l file prova che è contenuto nella cartella corso .

Ci sono alcuni attributi che possono essere utili per il controllo dei colori dei link di testo.

Questa pagina ti insegnerà a:

- Definire i colori per tutti i link sulla pagina.

Definire i colori per tutti i link sulla pagina

Il colore generale dei link di testo è specificato nel tag `<body>` , come nell'esempio qui sotto:

```
<body link="#C0C0C0" vlink="#808080" alink="#FF0000">
```

link - link normale - per una pagina dove il visitatore non è ancora stato. (il colore standard è blu - #0000FF).

· **vlink** - link visitato - per una pagina dove il visitatore è già stato. (il colore standard è viola - #800080).

· **alink** - link attivo - il colore del link quando è toccato dal mouse. (il colore standard è rosso - #FF0000).

LINK INTERNO

Creare dei link per le ancore ("anchors") è molto simile a creare link normali. Normalmente i link puntano sull'inizio di una pagina. Le ancore puntano invece su una zona all'interno di una pagina.

Il simbolo # davanti alla locazione di un link specifica che il link punta sull'ancora di una pagina. (Per ancora si intende un luogo specifico al centro della pagina).

Per fare un link ad un'ancora devi:

- **Creare un link che punti sull'ancora**
- **Creare l'ancora vera e propria.**

L'ancora viene creata usando il tag `<a>` .

Se vuoi creare un'ancora chiamata **capitolo4** , devi soltanto inserire questa riga nel punto in cui vuoi mettere l'ancora:

```
<a name="capitolo4"></a>
```

Dopo aver fatto questo, puoi creare un link puntato sull'ancora usando normalmente il tag `<a href>` , in questa maniera:

Clicca `qui` per leggere il capitolo 4.

Nota:

Ricordati di mettere il simbolo # davanti all'ancora.

Immagini HTML :

GIF E JPG

I computer immagazzinano le immagini in diverse maniere.

A lcuni metodi si basano sulla massima compressione possibile dell'immagine.

U no dei maggiori problemi circa l'uso delle immagini nei siti web consiste nel fatto che scaricare le immagini richiede molto più tempo rispetto al testo.

Per ridurre il più possibile il tempo necessario a scaricare, vengono usati dei formati di compressione di immagini. I due migliori usati sul web sono:

GIF**256 colori****Può gestire aree trasparenti****Questo formato non va bene per comprimere fotografie****In generale, è eccellente per titoli, bottoni e clipart****JPG****Colori illimitati****Non può gestire aree trasparenti****Eccellente nella compressione di fotografie e immagini complesse****In generale, non è adatto per titoli, bottoni e clipart.**

Questo significa che:

- Titoli, bottoni, divisori, clipart e altre semplici immagini funzionano meglio in **GIF**.
- Fotografie e immagini complesse normalmente funzionano meglio in **JPG**.

Se vuoi usare un'immagine che è in formato diverso da JPG o GIF, devi scaricare l'immagine in un programma di grafica e salvarla sia come JPG che come GIF.

INSERIMENTO IN HTML

Il tag usato per inserire un'immagine si chiama **img**.

Qui sotto puoi vedere un'immagine chiamata "cale.gif".

Ecco il codice HTML usato per inserire l'immagine in questa pagina web:

```

```

Se l'immagine viene immagazzinata nello stesso folder della pagina HTML, si può tralasciare il riferimento al dominio (in questo caso, www.echoecho.com/) e passare direttamente a inserire l'immagine con questo codice:

```

```

Nelle pagine seguenti tratteremo i diversi modi di controllare come l'immagine viene inserita.

RIDIMENSIONAMENTO

Puoi cambiare le dimensioni di un'immagine usando gli attributi **width** e **height** (larghezza e altezza).

In generale, non è consigliabile ridurre le dimensioni delle immagini usando questi comandi, in quanto l'immagine viene trasferita in internet nella sua misura originale, indipendentemente dalla riduzione che si è operata su di essa. Questo rallenterebbe lo scaricamento della tua pagina web.

Questo significa che se hai un'immagine più grande di quanto la vorresti, per inserirla in dimensioni ridotte sulla tua pagina, dovrai ridurla in un programma di grafica, invece di ridurle semplicemente le dimensioni sulla pagina usando gli attributi width e height.

Al contrario, a volte, può essere vantaggioso ingrandire le immagini usando questa tecnica.

Ecco due presentazioni della stessa identica immagine - con differenti opzioni per larghezza (width) e altezza (height).



```

```



```

```

Se non vengono specificate le opzioni di larghezza e altezza, il browser automaticamente userà le dimensioni reali dell'immagine.

Comunque, dovresti sempre specificare le opzioni di larghezza e altezza, anche quando usi un'immagine nelle sue dimensioni reali!

Il motivo è che se queste opzioni vengono tralasciate, il browser non può costruire la pagina finché l'immagine non è completamente scaricata.

Questo vuol dire che il visitatore non può leggere il testo intorno all'immagine mentre l'immagine stessa viene scaricata - e questo dà come risultato l'impressione di una pagina "lenta".

Questo è ancora più evidente se l'immagine è all'interno di una tavola.

In tal caso, l'intera tavola appare solo quando l'immagine è completamente scaricata.

BORDO INTORNO

Puoi aggiungere un bordo all'immagine usando l'opzione **border** come puoi vedere nell'esempio qui sotto:

Nota:

I browser Netscape mostreranno il bordo soltanto se l'immagine è un link.



```

```

L'aggiunta di un bordo all'immagine serve anche per aiutare il visitatore a capire immediatamente che l'immagine in questione è un link. Comunque, la rete è piena di immagini - link che non hanno bordi, così normalmente i visitatori fanno scorrere il mouse sulle immagini per vedere se queste sono anche link.

Comunque - se hai un'immagine che crea spesso dubbi, dovresti considerare l'ipotesi di aggiungerle un bordo - o forse faresti addirittura meglio a cambiare del tutto l'immagine - perché se questa non mostra chiaramente di essere un link, vuol dire che non è efficace.

TESTO ALTERNATIVO

Puoi aggiungere un testo alternativo a un'immagine usando l'opzione **alt**, come puoi vedere qui sotto:



```

```

Bisognerebbe sempre aggiungere testi alternativi alle immagini, in modo che gli utenti possano farsi un'idea dell'immagine prima che questa venga scaricata.

Questo è particolarmente importante se l'immagine è un link.

Ci sono poche cose che irritano di più di voler lasciare la pagina corrente - e allo stesso tempo essere obbligati ad aspettare che un'immagine venga completamente scaricata prima di poterlo fare.

Viene allora quasi spontaneo adoperare i semplici comandi del browser e abbandonare l'intero sito.

ALLINEAMENTO

Si possono allineare le immagini in conformità con il testo che le circonda, usando i seguenti allineamenti:

- **default** allinea l'immagine usando gli attributi di default del Web browser. E' come baseline.
- **left** allinea l'immagine al margine sinistro e avvolge il testo che segue l'immagine.
- **right** allinea l'immagine al margine destro e avvolge il testo che precede l'immagine.
- **top** allinea la parte superiore dell'immagine con il testo intorno.
- **texttop** allinea la parte superiore dell'immagine con la parte superiore del testo situato più in alto sulla linea.
- **middle** allinea la parte mediana dell'immagine con il testo intorno.
- **absmiddle** allinea l'immagine con il centro della linea corrente.
- **baseline** allinea l'immagine con il baseline della linea corrente.
- **bottom** allinea la parte inferiore dell'immagine con il testo intorno.
- **absbottom** allinea l'immagine con la parte inferiore della linea corrente.
- **center** allinea il centro dell'immagine con il testo intorno.

Le tabelle (tables) vengono adoperate nei siti per due ragioni principali

- La prima, più ovvia, è quella di sistemare le informazioni in una tabella.
- La seconda - meno ovvia - è quella di creare il layout della pagina servendosi di tabelle nascoste.

L'uso delle tabelle per dividere la pagina in diverse sezioni è uno strumento estremamente efficace. Quasi tutti i siti più grandi sulla rete si servono di tabelle nascoste per strutturare le pagine.

Gli aspetti più importanti della progettazione di una pagina che si possono trattare con l'uso di tabelle sono:

- La divisione della pagina in sezioni separate.

Una tabella invisibile è una risorsa eccellente per questo scopo.

- La creazione di menù.

In genere con un colore per la voce principale e un altro per i link che seguono nelle righe successive.

- L'inserimento di form field interattivi.

Generalmente un'area grigia contenente un'opzione di ricerca.

- La creazione di titoli di pagina rapidamente scaricabili.

Una tabella colorata con un testo viene scaricata in un batter d'occhio rispetto ad un banner anche semplicissimo.

- L'allineamento di immagini che sono state tagliate in parti più piccole.

- La disposizione del testo in due o più colonne sistemate una accanto all'altra.

L'importanza delle tabelle nella progettazione non deve essere tuttavia sopravvalutata. Ci sono alcune cose da tenere presenti quando si decide di farne uso.

La più importante è che il contenuto della tabella viene mostrato solo quando l'intera tabella è scaricata. Se hai pagine molto lunghe, ti consigliamo di dividerle in più tabelle - in modo che l'utente possa leggerne l'inizio, mentre il resto della pagina viene scaricato.

Ora andiamo avanti per esaminare le diverse tecniche...

Tabelle HTML : TAG FONDAMENTALI

Le tabelle sono definite con il tag `<table>` .

Per inserire una tabella sulla tua pagina devi semplicemente aggiungere questi tag nel punto in cui vuoi metterla.

```
<table>
</table>
```

La tabella qui sopra non può funzionare, perché non ha né righe né colonne.

ROWS:

Per inserire le righe nella tua tabella, adopera i tag `<tr>` e `</tr>` .

Esempio:

```
<table>
<tr>Questa è la prima riga.</tr>
<tr>Questa è la seconda riga.</tr>
</table>
```

Risultato:

Questa è la prima riga.
Questa è la seconda riga.

COLONNE:

Puoi dividere le righe in colonne con i tag `<td>` e `</td>` :

Esempio:

```
<table>
<tr> <td>Prima riga, sinistra.</td> <td>Prima riga, destra.</td> </tr>
<tr> <td>Seconda riga, sinistra.</td> <td>Seconda riga, destra.</td> </tr>
</table>
```

Risultato:

Prima riga, sinistra. Prima riga, destra.
Seconda riga, sinistra. Seconda riga, destra.

Questa pagina ti ha mostrato i punti essenziali delle tabelle. Devi sapere che esistono diverse opzioni per personalizzare le tue tabelle.

Le prossime pagine sono dedicate ai diversi attributi per i tag `<table>` , `<tr>` e `<td>` .

Tabelle HTML : TAG PER LE TABELLE

Le seguenti proprietà possono essere aggiunte al tag `<table>` :

| Proprietà | Descrizione |
|---|--|
| <code>align=</code> <code>left</code> <code>center</code> <code>right</code> | Tabella allineata a sinistra Tabella al centro Tabella allineata a destra |
| <code>background=nomefile.gif o .jpg</code> | immagine inserita dietro alla tabella |
| <code>bgcolor=#rrggbb</code> | colore dello sfondo |
| <code>border=n</code> | spessore del bordo |
| <code>bordercolor=#rrggbb</code> | colore del bordo |
| <code>bordercolordark=#rrggbb</code> | ombra del bordo |
| <code>cellpadding=n</code> | spazio tra le celle |
| <code>cellspacing=n</code> | distanza tra cella e contenuto |
| <code>nowrap</code> | impedisce l'interruzione di riga, anche se il contenuto è più grande della finestra del browser. |
| <code>valign=</code> <code>top</code> <code>bottom</code> | allinea il contenuto con il lato superiore della cella allinea il contenuto con il lato inferiore della cella |
| <code>width=</code> <code>n,n</code> <code>n,n%</code> | ampiezza minima della tabella in pixel ampiezza minima in percentuale rispetto alla misura della finestra |

Nota:

Le proprietà della tabella si intendono per l'intera tabella.

Se determinate proprietà sono destinate alle singole celle, avranno priorità rispetto agli attributi fissati per l'intera pagina.

Tabelle HTML : I TAG ROW/CELL

Questi attributi si possono aggiungere ai tag `<tr>` e `<td>` .

| PROPRIETA' | DESCRIZIONE |
|--|--|
| <code>align=</code> <code>left</code> <code>right</code> <code>center</code> | allinea il contenuto a sinistra nelle celle allinea il contenuto a destra nelle celle allinea il contenuto al centro delle celle |
| <code>background=nomefile.gif o .jpg</code> | definisce un'immagine di sfondo per le celle |
| <code>bgcolor=#rrggbb</code> | definisce un colore di sfondo per le celle |
| <code>bordercolor=#rrggbb</code> | definisce il colore del bordo delle celle |
| <code>bordercolordark=#rrggbb</code> | definisce il colore per l'ombra del bordo delle celle |
| <code>valign=</code> <code>top</code> <code>middle</code> <code>bottom</code> | allinea al lato superiore delle celle allinea al centro delle celle allinea al lato inferiore delle celle |
| <code>width=</code> <code>n</code> <code>n%</code> | specifica l'ampiezza minima delle celle in pixel specifica l'ampiezza minima delle celle in percentuale rispetto all'ampiezza della tabella |
| <code>height=</code> <code>n</code> <code>n%</code> | altezza minima delle celle in pixel altezza minima delle celle in percentuale rispetto all'altezza della tabella |

Questi attributi sono validi soltanto per i tag `<td>` .

PROPRIETA' DESCRIZIONE

| | |
|------------------------|--|
| <code>colspan=n</code> | numero delle colonne nella cella |
| <code>nowrap</code> | impedisce l'interruzione di riga, anche se il contenuto della cella è più grande della finestra del browser. |
| <code>rowspan=n</code> | numero delle righe nella cella |

Nota:

Gli attributi per le colonne (tag `<td>`) hanno priorità rispetto a quelli per le righe (tag `<tr>`).

Gli attributi per le celle (tag `<tr>` o `<td>`) hanno priorità rispetto agli attributi per l'intera tabella (tag `<table>`).

INTRODUZIONE

I frame ("cornici") possono dividere lo schermo in finestre separate.

Ognuna di queste finestre può contenere un documento in HTML.

Un file che specifica il modo in cui lo schermo è diviso si chiama frameset.

Se vuoi creare una pagina con dei frame, ecco cosa devi fare:

- creare un documento HTML con il frameset
- creare i normali documenti HTML che dovranno essere contenuti in ognuno di questi frame.

Quando una pagina frameset viene scaricata, il browser automaticamente scarica ognuna delle pagine associate ai frame.

ESEMPIO BASE

Un frameset è in poche parole un documento HTML che dice al browser come dividere lo schermo in finestre separate.

Ecco l'HTML per il frameset qui sopra::

```
<html>
<head>
<title>La Mia Pagina Frame</title>
</head>

<frameset cols="120,*">
<frame src="menupage.htm" name="menu">
<frameset rows="*,50">
<frame src="welcomepage.htm" name="main">
<frame src="bottombanner.htm" name="bottom">
</frameset>
</frameset>

</html>
```

Hai notato? Il frameset è di sole sette righe!

Dividiamolo e osserviamo le righe una ad una...

LA CREAZIONE DI UN FRAMESET

Come abbiamo detto nella pagina precedente, un frameset è semplicemente un documento HTML che indica al browser come dividere lo schermo in finestre separate.

Se il frameset fosse così:

```
Il suo codice sarebbe:
<frameset cols="120,*">
</frameset>
```

Lo schermo è diviso in due colonne.

Quella a sinistra è di 120 pixel e quella di destra occupa il resto dello schermo (indicato con *).

Se le finestre non avessero un nome, sarebbe impossibile usare il frameset.

Andiamo avanti e impariamo ad aggiungere nomi e pagine default alle finestre frame.

PAGINE DI DEFAULT

Puoi aggiungere pagine default alle finestre frame con l'attributo **src**.

Le pagine default sono le pagine che verranno scaricate quando il frameset verrà aperto per la prima volta.

Possiamo ulteriormente aggiungere nomi ad ognuna delle finestre frame adoperando l'attributo **name**.

Questo ci permetterà di creare un link in una finestra frame, aprire una pagina in un'altra finestra frame.

In questo esempio abbiamo aggiunto nomi e pagine default alle finestre frame:

```
<frameset cols="120,*" >
<frame src="menu.htm" name="menu" >
<frame src="frontf.htm" name="principale" >
</frameset>
```

Ecco come apparirà l'intero frameset:

m
e
n principale
u

A bbiamo ancora diviso lo schermo in due colonne, quella di sinistra misura 120 pixel, mentre quella di destra occupa il resto dello schermo. (Alcuni schermi misurano in larghezza 640 pixel, altri 800 e alcuni fino a 1024, ecco perché c'è bisogno del *).

M a stavolta abbiamo anche detto al browser che la finestra frame di sinistra deve contenere una pagina HTML chiamata **menu.htm** e che la colonna a destra deve contenere un documento HTML chiamato **frontf.htm** .

I n più abbiamo assegnato i nomi "menu" e "principale" alle due finestre frame, così adesso siamo anche in grado di creare link per finestre specifiche.

A bbiamo chiamato i frame **menu** e **principale** , ma sarebbe stato possibile dargli qualsiasi altro nome.

I l tipo di impaginazione con una finestra di menù a sinistra e una finestra principale a destra è il frameset più comune che si può incontrare sulla rete.

C i sono alcuni altri attributi che potremmo aggiungere al frameset.

P er esempio potresti voler rendere invisibili i bordi della cornice.

BORDI

P er rendere invisibili i bordi devi semplicemente aggiungere i parametri "cols-line" al frameset :

```
<frameset cols="120,*" frameborder="0" border="0" framespacing="0" >
<frame src="menu.htm" name="menu" >
<frame src="frontf.htm" name="principale" >
</frameset>
```

O ra l'intero frameset apparirà così:

m
e
n
u

principale

P ossiamo aggiungere ulteriori parametri al nostro frameset...

FINESTRE RIDIMENSIONABILI

S e vuoi che la grandezza delle finestre non sia modificabile, devi soltanto aggiungere il parametro "noresize" al frame src lines:

```
<frameset cols="120,*" frameborder="0" border="0" framespacing="0">
<frame src="menu.htm" name="menu" noresize >
<frame src="frontf.htm" name="principale" noresize >
</frameset>
```

BARRE DI SCORRIMENTO

P oniamo che tu non voglia lo scroll bar (la barra - cursore di scorrimento pagina) nella finestra del menu.

I noltre la finestra principale dovrebbe avere la barra di scorrimento solo quando ciò fosse necessario (se il documento HTML non entrasse tutto nella finestra), ma se non ce ne fosse bisogno - non dovrebbero esserci barre di scorrimento.

I l codice allora sarebbe questo:

```
<frameset cols="120,*" frameborder="0" border="0" framespacing="0">
<frame src="menu.htm" name="menu" noresize scrolling=no >
<frame src="frontf.htm" name="principale" noresize scrolling=auto >
</frameset>
```

T ralasciando di specificare l'attributo per lo scorrimento della pagina, la barra apparirà soltanto in caso di necessità - altrimenti no.

LINKS ALL'INTERNO

S e hai un documento HTML con un link sul testo "Analisi" per esempio, che porta a una pagina chiamata "analisi.htm", questo apparirà nel documento come:

Vai all' [Analisi](analisi.htm) del progetto

O ra, se il link fosse nella finestra del menu del nostro esempio, e volessimo che questo caricasse una pagina nella finestra principale, il codice HTML sarebbe:

Vai all' [Analisi](analisi.htm) del progetto

A bbiamo semplicemente aggiunto il parametro **target="principale"** al tag **<a href>** .

O ra il link si aprirà sulla finestra frame principale invece che su quella del menù, dove si trova il link stesso.

Q uattro nomi di target sono riservati, e il browser li interpreterà in questo modo:

- **_blank** scarica la pagina in una nuova finestra browser
- **_self** scarica la pagina nella finestra corrente.
- **_parent** scarica la pagina nel frame superiore a quello in cui si trova l'hyperlink .
- **_top** cancella tutti i frames, scarica la finestra browser al completo.

L eggendo le pagine di questa sezione, imparerai tutto quello che c'è da sapere sui frameset in HTML.

E ' possibile perfezionare il controllo dei tuoi frameset con **javascript** . Per esempio, javascript può impedire alle pagine progettate per funzionare solo all'interno di frameset, di venire scaricate fuori dal frameset.

ESEMPI

I n questa pagina puoi vedere alcuni esempi di diversi frameset:

top

bottom

```
<frameset rows="16%,84%">
<frame src="top.htm" name="top">
<frame src="bottom.htm" name="bottom">
</frameset>
```

tl tr

bottom

```
<frameset rows="16%,84%">
<frameset cols="50%,50%">
<frame src="tl.htm" name="tl">
<frame src="tr.htm" name="tr">
</frameset>
<frame src="bottom.htm" name="bottom">
</frameset>
```

top

left right

```
<frameset rows="16%,84%">
<frame src="top.htm" name="top">
<frameset cols="50%,50%">
<frame src="left.htm" name="left">
<frame src="right.htm" name="right">
</frameset>
</frameset>
```

topleft topright

botleft botright

```
<frameset rows="50%,50%" cols="50%,50%">
<frame src="topleft.htm" name="topleft">
<frame src="topright.htm" name="topright">
<frame src="botleft.htm" name="botleft">
<frame src="botright.htm" name="botright">
</frameset>
```

topleft topright

botleft brtl brtr
botrbot

```
<frameset rows="50%,50%" cols="50%,50%">
<frame src="topleft.htm" name="topleft">
<frame src="topright.htm" name="topright">
<frame src="botleft.htm" name="botleft">
<frameset rows="50%,50%">
<frameset cols="50%,50%">
<frame src="brtl.htm" name="brtl">
<frame src="brtr.htm" name="brtr">
</frameset>
<frame src="botrbot.htm" name="botrbot">
</frameset>
</frameset>
```

topleft topright

botleft botright

```
<frameset rows="240,240" cols="320,320">
<frame src="topleft.htm" name="topleft">
<frame src="topright.htm" name="topright">
<frame src="botleft.htm" name="botleft">
<frame src="botright.htm" name="botright">
</frameset>
```

topleft topright

botleft botright

```
<frameset rows="50%,*" cols="320,*">
<frame src="topleft.htm" name="topleft">
<frame src="topright.htm" name="topright">
<frame src="botleft.htm" name="botleft">
<frame src="botright.htm" name="botright">
</frameset>
```

| | | |
|-----------------------------|----------------------------------|---|
| &#160; | Non-breaking space | |
| &#161; &#iexcl; | Inverted exclamation | ¡ |
| &#162; &#cent; | Cent sign | ¢ |
| &#163; &#pound; | Pound sterling sign | £ |
| &#164; &#curren; | General currency sign | ¤ |
| &#165; &#yen; | Yen sign | ¥ |
| &#166; &#brvbar; | Broken vertical bar | ¸ |
| &#167; &#sect; | Section sign | § |
| &#168; &#uml; | Umlaut (dierisis) | ¨ |
| &#169; &#copy; | Copyright | © |
| &#170; &#ordf; | Feminine ordinal | ª |
| &#171; &#laquo; | Left angle quote, guillemot left | « |
| &#172; &#not; | Not sign | ¬ |
| &#173; &#shy; | Soft hyphen | - |
| &#174; &#reg; | Registered trademark | ® |
| &#175; &#macr; | Macron accent | - |
| &#176; &#deg; | Degree sign | ° |
| &#177; &#plusmn; | Plus or minus | ± |

| | | |
|-----------------|-------------------------------------|---|
| ² ² | Superscript two | ² |
| ³ ³ | Superscript three | ³ |
| ´ ´ | Acute accent | ´ |
| µ µ | Micro sign | µ |
| ¶ ¶ | Paragraph sign | ¶ |
| · · | Middle dot | · |
| ¸ ¸ | Cedilla | ¸ |
| ¹ ¹ | Superscript one | ¹ |
| º º | Masculine ordinal | ¼ |
| » » | Right angle quote, guillemot right | » |
| ¼ ¼ | Fraction (one quarter) | ¼ |
| ½ ½ | Fraction (one half) | ½ |
| ¾ ¾ | Fraction (three quarters) | ¾ |
| ¿ ¿ | Inverted question mark | ¿ |
| À À | Capital A, grave accent | À |
| Á Á | Capital A, acute accent | Á |
| Â Â | Capital A, circumflex accent | Â |
| Ã Ã | Capital A, tilde | Ã |
| Ä Ä | Capital A, umlaut (dierisis) | Ä |
| Å Å | Capital A, ring | Å |
| Æ Æ | Capital AE diphthong (ligature) | Æ |
| Ç Ç | Capital C, cedilla | Ç |
| È È | Capital E, grave accent | È |
| É É | Capital E, acute accent | É |
| Ê &Eirc; | Capital E, circumflex accent | Ê |
| Ë Ë | Capital E, umlaut (dierisis) | Ë |
| Ì Ì | Capital I, grave accent | Ì |
| Í Í | Capital I, acute accent | Í |
| Î Î | Capital I, circumflex accent | Î |
| Ï Ï | Capital I, umlaut (dierisis) | Ï |
| Ð Ð | Capital Eth, Icelandic | Ð |
| Ñ Ñ | Capital N, tilde | Ñ |
| Ò Ò | Capital O, grave accent | Ò |
| Ó Ó | Capital O, acute accent | Ó |
| Ô Ô | Capital O, circumflex accent | Ô |
| Õ Õ | Capital O, tilde | Õ |
| Ö Ö | Capital O, umlaut (dierisis) | Ö |
| × × | Multiply sign | × |
| Ø Ø | Capital O, slash | Ø |
| Ù Ù | Capital U, grave accent | Ù |
| Ú Ú | Capital U, acute accent | Ú |
| Û Û | Capital U, circumflex accent | Û |
| Ü Ü | Capital U, umlaut (dierisis) | Ü |
| Ý Ý | Capital Y, acute accent | Ý |
| Þ Þ | Capital Thorn, Icelandic | Þ |
| ß ß | Small sharp s, German (sz ligature) | ß |
| à à | Small a, grave accent | à |
| á á | Small a, acute accent | á |
| â â | Small a, circumflex accent | â |
| ã ã | Small a, tilde | ã |
| ä ä | Small a, umlaut (dierisis) | ä |
| å å | Small a, ring | å |
| æ æ | Small ae diphthong (ligature) | æ |
| ç ç | Small c, cedilla | ç |
| è è | Small e, grave accent | è |
| é é | Small e, acute accent | é |
| ê ê | Small e, circumflex accent | ê |
| ë ë | Small e, umlaut (dierisis) | ë |
| ì ì | Small i, grave accent | ì |
| í í | Small i, acute accent | í |

| | | |
|-----------------|----------------------------|---|
| î î | Small i, circumflex accent | î |
| ï ï | Small i, umlaut (dierisis) | ï |
| ð ð | Small eth, Icelandic | ð |
| ñ ñ | Small n, tilde | ñ |
| ò ò | Small o, grave accent | ò |
| ó ó | Small o, acute accent | ó |
| ô ô | Small o, circumflex accent | ô |
| õ õ | Small o, tilde | õ |
| ö ö | Small o, umlaut (dierisis) | ö |
| ÷ ÷ | Division sign | ÷ |
| ø ø | Small o, slash | ø |
| ù ù | Small u, grave accent | ù |
| ú ú | Small u, acute accent | ú |
| û û | Small u, circumflex accent | û |
| ü ü | Small u, umlaut (dierisis) | ü |
| ý ý | Small y, acute accent | ý |
| þ þ | Small thorn, Icelandic | þ |
| ÿ ÿ | Small y, umlaut (dierisis) | ÿ |

Caratteri supportati da Explorer – non supportati da Netscape

| NUMBER NAME | DESCRIPTION | EX |
|-------------|----------------------------------|----|
| ‚ | Comma | |
| ƒ | Mathematical function f | |
| „ | Low quotation mark | |
| … | Triple dots | |
| † | Footnote mark | |
| ‡ | Footnote mark (2) | |
| ˆ | Circumflex accent | |
| ‰ | Perthousandth | |
| Š | Capital S, inverted circumflex | |
| ‹ | Small less than | |
| Œ | Capital OE, diphthong (ligature) | |
| ‘ | Single quote (opening) | ‘ |
| ’ | Single quote (closing) | ’ |
| “ | Double quote (opening) | |
| ” | Double quote (closing) | |
| • | Large middle dot | • |
| – | Middle dash | – |
| — | Double middle dash | — |
| ˜ | Tilde | |
| ™ | (See Note!) Trademark | |
| š | Small s, inverted circumflex | |
| › | Small greater than | |
| œ | Small oe, diphthong (ligature) | |
| Ÿ | Capital Y, umlaut (dierisis) | |

IL TAG FORM

Quando viene lanciato un form, tutti i suoi campi vengono inviati.

Il tag `<form>` indica al browser l'inizio e la fine del form. Puoi aggiungere tutti i tipi di tag HTML tra i tag `<form>` e `</form>`.

Questo significa che un form può agevolmente includere una tavola o un'immagine insieme ai form field.

Guarda questo esempio:

```
<html>
<head>
<title>La mia pagina</title>
</head>
```

```

<body>
<!-- Qui va HTML -->
<form>
<!-- Qui vanno form field e HTML -->
</form>
<!-- Qui va HTML -->
</body>
</html>

```

Nota:

Diversamente da una table, i form non sono visibili sulla pagina.

Il form nel nostro esempio è inutile :

- Innanzitutto non contiene form field. E' paragonabile ad un semplice foglio di carta bianco.
- In secondo luogo, non contiene un ricevente per il form.

Per fare sì che il browser sappia dove inviare il contenuto, dobbiamo aggiungere questi attributi al tag **<form>** :

- **action= address**
- **method= post** oppure **method= get**

L' **address** è l'url del programma cgi a cui il contenuto deve essere inviato.

I metodi **post** e **get** sono semplicemente due metodi differenti di mandare dati al programma.

Qui sotto puoi vedere un esempio di un tipico form tag, con azione e metodo specificati.

```

<html>
<head>
<title>La mia pagina</title>
</head>

<body>
<!-- Qui va HTML -->
<form method="post" action="http://www.ecn.org/cgi-bin/formmail.cgi">
<!-- Qui vanno form field e HTML -->
</form>
<!-- Qui va HTML -->
</body>
</html>

```

FORM FIELDS

Questi field possono essere aggiunti ai tuoi form:

- **Text field**
- **Password field**
- **Hidden field**
- **Text area**
- **Check box**
- **Radio button**
- **Drop-down menu**
- **Submit button**
- **Reset button**
- **Image button**

FIELD TESTO

I field di testo (text fields) sono aree di una riga sola che permettono all'utente di inserire testo.

Se hai bisogno di più righe, devi usare un **text area**.

ATTRIBUTI:

Ecco una lista di attributi validi per i text field:

Inizio modulo

| HTML | SPIEGAZIONE | ESEMPIO |
|-------------------|------------------------------|----------------------|
| text | Text field di una riga | |
| size= | Caratteri mostrati. | |
| maxlength= | Numero massimo di caratteri. | |
| name= | Nome del field. | <input type="text"/> |
| value= | Valore iniziale nel field | |
| align= | Allineamento del field. | |
| tabindex= | Tab order del field. | |

L'opzione **size** definisce la larghezza del field. Da ciò deriva la quantità dei caratteri visibili che il field riesce a contenere.

L'opzione **maxlength** definisce la lunghezza massima del field. Da ciò deriva la quantità di caratteri che possono entrare nel field.

Se non viene specificata una lunghezza massima ("maxlength"), il visitatore può facilmente inserire più caratteri di quanti se ne possano visualizzare nel campo.

L'attributo **name** aggiunge un nome interno al field, così il programma che decifra il form può identificare i field.

L'attributo **value** definisce cosa apparirà nel box come valore di default.

L'attributo **align** definisce l'allineamento del field.

Entrate valide sono: TOP, MIDDLE, BOTTOM, RIGHT, LEFT, TEXTTOP, BASELINE, ABSMIDDLE, ABSBOTTOM. Gli allineamenti sono spiegati nella sezione immagine. Puoi saperne di più sui diversi allineamenti cliccando **qui**.

L'attributo **tabindex** stabilisce l'ordine in cui i diversi fields devono essere attivati quando il visitatore clicca sul tasto tab.

Inizio modulo

U N ESEMPIO:

Guarda questo esempio di HTML:

```
<html>
<head>
<title>La mia pagina</title>
</head>
<body>
<form name="myform" action="http://www.mydomain.com/myformhandler.cgi" method="POST">
<div align="center">
<br><br>
<input type="text" size="25" value="Scrivi qui il tuo nome!">
<br><br>
</div>
</form>
</body>
</html>
```

Ecco il risultato:

Inizio modulo



Fine modulo

PASSWORD

I field di password ("parola d'ordine") sono simili a quelli di testo.

La differenza è che ciò che viene inserito in un field di password viene mostrato sotto forma di puntini sullo schermo. Questo serve, naturalmente, a evitare che altri possano leggere la password sullo schermo.

I field di password ("parola d'ordine") sono simili a quelli di testo.

La differenza è che ciò che viene inserito in un field di password viene mostrato sotto forma di puntini sullo schermo. Questo serve, naturalmente, a evitare che altri possano leggere la password sullo schermo.

I field di password ("parola d'ordine") sono simili a quelli di testo.

La differenza è che ciò che viene inserito in un field di password viene mostrato sotto forma di puntini sullo schermo. Questo serve, naturalmente, a evitare che altri possano leggere la password sullo schermo.

I field di password ("parola d'ordine") sono simili a quelli di testo.

La differenza è che ciò che viene inserito in un field di password viene mostrato sotto forma di puntini sullo schermo. Questo serve, naturalmente, a evitare che altri possano leggere la password sullo schermo.

I field di password ("parola d'ordine") sono simili a quelli di testo.

La differenza è che ciò che viene inserito in un field di password viene mostrato sotto forma di puntini sullo schermo. Questo serve, naturalmente, a evitare che altri possano leggere la password sullo schermo.

I field di password ("parola d'ordine") sono simili a quelli di testo.

La differenza è che ciò che viene inserito in un field di password viene mostrato sotto forma di puntini sullo schermo. Questo serve, naturalmente, a evitare che altri possano leggere la password sullo schermo.

I field di password ("parola d'ordine") sono simili a quelli di testo.

La differenza è che ciò che viene inserito in un field di password viene mostrato sotto forma di puntini sullo schermo. Questo serve, naturalmente, a evitare che altri possano leggere la password sullo schermo.

I field di password ("parola d'ordine") sono simili a quelli di testo.

La differenza è che ciò che viene inserito in un field di password viene mostrato sotto forma di puntini sullo schermo. Questo

serve, naturalmente, a evitare che altri possano leggere la password sullo schermo. I field di password ("parola d'ordine") sono simili a quelli di testo.


L a differenza è che ciò che viene inserito in un field di password viene mostrato sotto forma di puntini sullo schermo. Questo serve, naturalmente, a evitare che altri possano leggere la password sullo schermo. I field di password ("parola d'ordine") sono simili a quelli di testo.

L a differenza è che ciò che viene inserito in un field di password viene mostrato sotto forma di puntini sullo schermo. Questo serve, naturalmente, a evitare che altri possano leggere la password sullo schermo.

A TTRIBUTI:

Qui sotto ecco una lista di attributi validi per i password field:

Inizio modulo

| HTML | SPIEGAZIONE | ESEMPIO |
|-------------------|------------------------------|---|
| password | Password field di una riga | |
| size= | Caratteri visibili. | |
| maxlength= | Numero massimo di caratteri. | |
| name= | Nome del field. |  |
| value= | Valore iniziale nel field. | |
| align= | Allineamento del field. | |
| tabindex= | Tab order del field. | |

L 'opzione **size** definisce la larghezza del field. Stabilisce quanti caratteri visibili può contenere.

L 'opzione **maxlength** definisce la lunghezza massima del field. Da ciò deriva la quantità di caratteri che possono essere inseriti nel field.

Se non viene specificata una lunghezza massima ("maxlength"), il visitatore può facilmente inserire più caratteri di quanti ne vengano mostrati sullo schermo.

L 'opzione **name** aggiunge un nome interno al field, in modo che il programma che gestisce il form possa identificare i field.

L 'attributo **value** definisce cosa apparirà nella casella come valore di default.

L 'attributo **align** definisce l'allineamento del field.

Le entrate valide sono: TOP, MIDDLE, BOTTOM, RIGHT, LEFT, TEXTTOP, BASELINE, ABSMIDDLE, ABSBOTTOM. Gli allineamenti sono spiegati nella sezione immagini. Se ti interessa questo argomento, clicca **qui** .

L 'attributo **tabindex** definisce in quale ordine i differenti field dovranno attivarsi quando il visitatore clicca sul tasto tab.

Inizio modulo

U N ESEMPIO:

Guarda questo esempio di HTML:

```
<html>
<head>
<title>La mia pagina</title>
</head>
<body>
<form name="myform" action="http://www.mydomain.com/myformhandler.cgi" method="POST">
<div align="center">
Inserire Password : <input type="password" size="25">
<br><br>
</div>
</form>
</body>
</html>
```

Ecco il risultato:

Inizio modulo

Inserire Password : 
Fine modulo

FIELD NASCOSTO

I field nascosti ("hidden fields") sono simili a field di testo, con una differenza importantissima!

L a differenza è che il field nascosto non è mostrato sulla pagina. Di conseguenza il visitatore non può scriverci nulla sopra; lo scopo di questo tipo di field è dunque:

Introdurre informazioni non accessibili al visitatore.

Inizio modulo

HTML SPIEGAZIONE ESEMPIO

hidden Hidden field
name= Nome del field.
value= Valore del field.

L 'attributo **name** aggiunge un nome interno al field, in modo che il programma che decifra il form possa identificare i fields.

L 'attributo **value** definisce cosa verrà mandato al momento dell'invio del form.

U N ESEMPIO:

G uarda questo esempio di HTML:

```
<html>
<head>
<title>La mia pagina</title>
</head>
<body>
<form name="myform" action="http://www.mydomain.com/myformhandler.cgi" method="POST">
<div align="center">
<input type="text" size="25" value="Scrivi qui il tuo nome!">
<input type="hidden" name="Language" value="Italian">
<br><br>
</div>
</form>
</body>
</html>
```

Inizio modulo

E cco il risultato:

Inizio modulo



Fine modulo

I l field nascosto non si vede, ma quando il form viene inviato, il field nascosto viene mandato insieme ad esso.

I n questo esempio il field nascosto indica al programma che decifra il form, che sarebbe preferibile che la lingua adoperata dall'utente fosse l'italiano.

AREA DI TESTO

I text areas (aree di testo) sono field di testo che possono occupare diverse righe.

D iversamente dalla maggior parte dei form field, le aree di testo non sono definite con un tag **<input>** .


D evi invece inserire un tag **<textarea>** nel punto in cui vuoi iniziare un'area di testo, e chiudere con un tag **</textarea>** nel punto in cui vuoi che l'area finisca.

T utte le cose scritte tra questi tag saranno inserite nel box "text area".

A TTRIBUTI:

E cco una lista di attributi validi per le aree di testo:

Inizio modulo

| HTML | SPIEGAZIONE | ESEMPIO |
|------------------|---|---|
| textarea | Area di testo - alcune righe | |
| rows= | Righe nel field. | |
| cols= | Colonne nel field. | |
| name= | Nome del field. | |
| tabindex= | Ordine tab del field. |  |
| wrap= | Ignora gli "a capo" | |
| off | Mostra gli "a capo", ma | |
| virtual | manda il testo come viene inserito. | |
| physical | Inserisce gli " a capo" quando è necessario e li trasmette. | |

Fine modulo

Gli attributi **cols** e **rows** sono diretti e semplici. Specificano il numero di colonne e righe che si vogliono inserire nell'area di testo.

L'attributo **name** aggiunge un nome interno al field, in modo che il programma che decifra il form sia in grado di identificare il field.

L'attributo **tabindex** definisce l'ordine di attivazione dei diversi field nel momento in cui il visitatore cliccherà sul tasto tab.

Le opzioni **wrap** sono la parte più complicata delle aree di testo.

Se scegli di mettere il wrap in posizione **off** il testo viene considerato come una lunga sequenza di testo senza gli "a capo".

Dando al wrap la qualità di **virtual** il testo appare sulla pagina come se riconoscesse le interruzioni degli "a capo" - ma quando il form viene inviato, le interruzioni non vengono attivate.

Dando al wrap la qualità di **physical** il testo viene mandato esattamente come appare sullo schermo, compresi gli "a capo".

UN ESEMPIO:

Guarda questo esempio HTML:

```
<html>
<head>
<title>La mia Pagina</title>
</head>

<body>
<form name="myform" action="http://www.mydomain.com/myformhandler.cgi" method="POST">
<div align="center">
Questo è fuori dall'area<br><br>
<textarea cols="40" rows="5" name="myname">
Ora siamo dentro l'area - che bello!.
</textarea>
<br><br>
E ora siamo di nuovo fuori dall'area.
</div>
</form>
</body>
</html>
```

Ecco il risultato:

Inizio modulo

Questo è fuori dall'area



E ora siamo di nuovo fuori dall'area

Fine modulo

CHECKBOX

Le caselle di scelta ("check boxes") si adoperano quando si vuole dare al visitatore la possibilità di selezionare una o più opzioni da una serie di alternative. Se si vuole permettere una sola opzione, bisogna allora usare i **radio button**.

ATTRIBUTI:

Ecco una lista di attributi validi per i check box:

Inizio modulo

| HTML | SPIEGAZIONE | ESEMPIO Scegli una o più opzioni |
|------------------|-------------------------------------|--|
| checkbox | Scegli una o più opzioni | |
| name= | Nome del field. | |
| value= | Valore inviato se selezionato. | <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> |
| align= | Allineamento del field. | |
| tabindex= | Ordine tab del field. | |
| checked | Selezione default per questo field. | |

L'attributo **name** aggiunge un nome interno al field, in modo che il programma che decifra il form sia in grado di identificare il field.

L'attributo **value** definisce cosa sarà inviato al momento della selezione.

L'attributo **align** definisce l'allineamento del field.

Le entrate valide sono: TOP, MIDDLE, BOTTOM, RIGHT, LEFT, TEXTTOP, BASELINE, ABSMIDDLE, ABSBOTTOM.

L'attributo **tabindex** definisce l'ordine in cui i diversi field si attiveranno quando il visitatore cliccherà sul tasto tab.

Inizio modulo

U N ESEMPIO:

Guarda questo esempio HTML:

```
<html>
<head>
<title>La mia Pagina</title>
</head>
<body>
<form name="myform" action="http://www.mydomain.com/myformhandler.cgi" method="POST">
<div align="center"><br>
<input type="checkbox" name="option1" value="cartine"> Cartine<br>
<input type="checkbox" name="option2" value="sigarette" checked> Sigarette<br>
<input type="checkbox" name="option3" value="gomme"> Gomme<br>
<br>
</div>
</form>
</body>
</html>
```

E d ecco il risultato:

Inizio modulo



Cartine



Sigarette



Gomme

Fine modulo

RADIO BUTTON

I radio buttons ("bottoni radio") vengono usati quando si vuole che il visitatore selezioni una - e soltanto una - opzione da una serie di alternative. Se si vogliono permettere più opzioni contemporaneamente, bisogna invece usare **check boxes** .

A TTRIBUTI:

E cco una lista degli attributi validi per i bottoni radio:

Inizio modulo

| HTML | SPIEGAZIONE | ESEMPIO |
|------------------|---------------------------------------|--|
| radio | Scegli una - e soltanto una - opzione | |
| name= | Nome del gruppo. | |
| value= | Valore inviato se prescelto. | <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> |
| align= | Allineamento del field. | |
| tabindex= | Ordine tab del field. | |
| checked | Controllo default del field. | |

L 'attributo **name** indica a quale gruppo di bottoni radio appartiene il field. Selezionando un bottone, tutti gli altri bottoni dello stesso gruppo non possono essere selezionati.

Se non si è in grado di definire a quale gruppo appartenga il bottone prescelto, si può soltanto avere un gruppo di bottoni radio su ogni pagina.

L 'attributo **value** definisce cosa verrà inviato al momento della scelta.

L 'attributo **align** definisce il tipo di allineamento del field.

Le entrate valide sono: TOP, MIDDLE, BOTTOM, RIGHT, LEFT, TEXTTOP, BASELINE, ABSMIDDLE, ABSBOTTOM.

L 'attributo **tabindex** definisca l'ordine in cui i diversi field verranno attivati nel momento in cui il visitatore cliccherà sul tasto tab.

Inizio modulo

U N ESEMPIO:

Guarda questo esempio HTML:

```
<html>
<head>
<title>La Mia Pagina</title>
</head>
<body>
<form name="myform" action="http://www.mydomain.com/myformhandler.cgi" method="POST">
```

```

<div align="center"><br>
<input type="radio" name="group1" value="Cartine"> Cartine <br>
<input type="radio" name="group1" value="Tabacco" checked> Tabacco <br>
<input type="radio" name="group1" value="Maschera Antigas"> Maschera Antigas
<hr>
<input type="radio" name="group2" value="Gelato"> Gelato <br>
<input type="radio" name="group2" value="Birra"> Birra<br>
<input type="radio" name="group2" value="Vino" checked> Vino <br>
</div>
</form>
</body>
</html>

```

Ecco il risultato:
Inizio modulo

Cartine
 Tabacco
 Maschera Antigas

Gelato
 Birra
 Vino

Fine modulo

COMBO

I drop-down menus ("menù a discesa") sono probabilmente gli oggetti più flessibili che puoi aggiungere ai tuoi form.

Il menù a discesa ha lo stesso scopo dei bottoni radio (una selezione soltanto) o dei "check boxes" (dove sono permesse selezioni multiple).

Il vantaggio del menù a discesa, se confrontato con i bottoni radio e i check box, è che occupa meno spazio. Ma questo è allo stesso tempo uno svantaggio, perché non si possono vedere immediatamente tutte le opzioni.

C'è un modo di aggirare il problema - con l'attributo "size", si può personalizzare il menù e rendere visibile più di un'opzione alla volta, ma facendo questo si perde il vantaggio di occupare meno spazio.

Insomma, per ogni decisione presa, c'è il rovescio della medaglia...

A alcune volte è meglio mettere dei menù a discesa al posto di field di testo, perché selezionare da un menù è più facile che scrivere. Altre volte ciò è vivamente consigliabile, ad esempio quando il programma che decifra il form non è in grado di interpretare un testo qualsiasi in entrata.

Per esempio, capita spesso di dover scegliere il proprio stato da un menù a discesa. Questo perché scegliere da un menù è più facile che scrivere il nome dello stato.

Nello stesso modo, può capitare spesso che ti venga chiesto di digitare le 2 lettere iniziali del tuo stato, sempre da un menù a discesa. Questo serve ad evitare confusione per il programma che decifra il form. Se, per esempio, il programma accettasse soltanto le maiuscole, il menù a discesa assicurerebbe la completa validità di tutte le entrate.

Un altro esempio tipico è quello di sostituire i link con menù a discesa.

A TTRIBUTI:

Ecco una lista di attributi validi per i menù a discesa:

Inizio modulo

| HTML | SPIEGAZIONE | ESEMPIO |
|------------------|---|--------------------------|
| select | Menù a discesa | |
| name= | Nome del field. | |
| size= | Voci visibili nella lista. | |
| multiple= | Permette scelte multiple se specificato. | <input type="checkbox"/> |
| option | Voci individuali nel menù. | |

selected **Seleziona la voce per default.**
value= **Valore da inviare se selezionato.**

I menù a discesa combinano `<select>` e `<option>` .
 Entrambi i tag hanno un tag di apertura e chiusura.

Un tipico esempio potrebbe essere:

```
<select>
<option>Latte</option>
<option>Caffè</option>
<option>Tè</option>
</select>
```

Il tag `<select>` definisce il menù

L'attributo **name** aggiunge un nome interno al field, in modo che il programma che decifra il form sia in grado di identificare il field.

L'opzione **size** stabilisce quante voci sono visibili nello stesso momento. Default è una delle voci.

L'attributo **multiple** , quando è presente, permette la scelta multipla.

Il tag `<option>` definisce le singole voci nel menù.

L'attributo **value** definisce cosa verrà inviato se la voce viene selezionata. Questo non è sempre la stessa cosa che compare sul menù. Se il nostro field fosse definito in questo modo:

```
<option value="ID"> Idaho </option>
```

nel menù comparirebbe "Idaho", ma quando il form viene inviato, soltanto la forma abbreviata "ID" verrebbe effettivamente mandata.

Puoi stabilire che una voce venga selezionata come default, aggiungendo l'opzione: `<option selected>`

UN ESEMPIO:

Guarda questo esempio HTML:

```
<html>
<head>
<title>La Mia Pagina</title>
</head>
<body>
<form name="myform" action="http://www.mydomain.com/myformhandler.cgi" method="POST">
<div align="center">
<select name="mydropdown">
<option value="Latte">Latte Fresco</option>
<option value="Formaggio">Formaggio Stagionato</option>
<option value="Pane">Pane Caldo</option>
</select>
</div>
</form>
</body>
</html>
```

Ecco il risultato:

Inizio modulo



Fine modulo

BOTTONE DI INVIO

Quando un visitatore clicca su un tasto d'invio, il form viene mandato all'indirizzo specificato nell'attributo **action** del tag `<form>` .

Sciacome i visitatori non sempre sono dei perfezionisti, dovresti considerare l'ipotesi di aggiungere una **convalida javascript** del contenuto prima che questo venga effettivamente inviato.

A TTRIBUTI:

Ecco una lista degli attributi validi per i bottoni di invio:

Inizio modulo

| HTML | SPIEGAZIONE | ESEMPIO |
|------------------|-----------------------------------|---------|
| submit | Bottone di invio | |
| name= | Nome del bottone. | |
| value= | Testo scritto sul bottone. | |
| align= | Allineamento del bottone. | |
| tabindex= | Ordine tab del bottone. | |

L'attributo **name** aggiunge un nome interno al bottone, in modo che il programma che decifra il form non confonda il bottone con gli altri field.

L'attributo **value** definisce cosa c'è scritto sul bottone.

L'attributo **align** definisce l'allineamento del bottone.

Le entrate valide sono: TOP, MIDDLE, BOTTOM, RIGHT, LEFT, TEXTTOP, BASELINE, ABSMIDDLE, ABSBOTTOM.

L'attributo **tabindex** definisce l'ordine in cui i differenti fields verranno attivati quando il visitatore cliccherà sul tasto tab.

UN ESEMPIO:

Guarda questo esempio HTML

```
<html>
<head>
<title>La Mia Pagina</title>
</head>
<body>
<form name="myform" action="http://www.mydomain.com/myformhandler.cgi" method="POST">
<div align="center">
<br><br>
<input type="text" size="25" value="Inserisci qui il tuo nome!">
<br><input type="submit" value="Mandami il tuo nome!"> <br>
</div>
</form>
</body>
</html>
```

Ecco il risultato::

Inizio modulo



BOTTONE RESET

Quando un visitatore clicca su un bottone di reset, le entrate sono resettate ai valori di default.

A TTRIBUTI:

Ecco una lista degli attributi validi per i bottoni di reset:

Inizio modulo

| HTML | SPIEGAZIONE | ESEMPIO |
|------------------|-----------------------------------|---------|
| reset | Bottone reset | |
| name= | Nome del bottone. | |
| value= | Testo scritto sul bottone. | |
| align= | Allineamento del bottone. | |
| tabindex= | Ordine tab del bottone. | |

L'attributo **name** aggiunge un nome interno al bottone, in modo che il programma che decifra il form non confonda il bottone con gli altri fields.

L'attributo **value** definisce cosa c'è scritto sul bottone.

L'attributo **align** definisce l'allineamento del bottone.

Le entrate valide sono: TOP, MIDDLE, BOTTOM, RIGHT, LEFT, TEXTTOP, BASELINE, ABSMIDDLE, ABSBOTTOM.

L'attributo **tabindex** definisce l'ordine in cui i differenti field verranno attivati quando il visitatore cliccherà sul tasto tab.

UN ESEMPIO:

Guarda questo esempio HTML:

```
<html>
<head>
<title>My Page</title>
</head>
<body>
<form name="myform" action="http://www.mydomain.com/myformhandler.cgi" method="POST">
<div align="center">
<br><br>
<input type="text" size="25" value="Scrivi qui il tuo nome!">
<br><input type="submit" value="Mandami il tuo nome!"> <input type="reset" value="Reset!"> <br>
</div>
</form>
</body>
</html>
```

Ecco il risultato:

Inizio modulo



Fine modulo


BOTTONE IMMAGINE

I bottoni di immagine hanno lo stesso effetto dei bottoni di invio. Quando un visitatore clicca su un bottone di immagine il form viene mandato all'indirizzo specificato nell'attributo **action** del tag **<form>**.

ATTRIBUTI:

Ecco una lista di attributi validi per i bottoni di immagine:

Inizio modulo

| HTML | SPIEGAZIONE | ESEMPIO |
|------------------|---|---|
| image | Bottone di invio | |
| name= | Nome dell'immagine. | |
| src= | Url dell'immagine. | |
| align= | Allineamento dell'immagine. | |
| border= | Spessore del bordo intorno all'immagine. |  |
| width= | Larghezza dell'immagine. | |
| height= | Altezza dell'immagine. | |
| vspace= | Spazio sopra e sotto l'immagine. | |
| hspace= | Spazio a destra e sinistra dell'immagine. | |
| tabindex= | Ordine tab dell'immagine. | |

L'attributo **name** aggiunge un nome interno al bottone, in modo che il programma che decifra il form non confonda il bottone con gli altri field.

L'attributo **src** definisce l'URL dell'immagine.

L'attributo **align** definisce l'allineamento dell'immagine.

Le entrate valide sono: TOP, MIDDLE, BOTTOM, RIGHT, LEFT, TEXTTOP, BASELINE, ABSMIDDLE, ABSBOTTOM.

L'attributo **border** definisce lo spessore (in pixel) del bordo intorno all'immagine.

L'attributo **width** definisce la larghezza dell'immagine.

L'attributo **height** definisce l'altezza dell'immagine.

L'attributo **vspace** definisce lo spazio sopra e sotto l'immagine.

L'attributo **hspace** definisce lo spazio a destra e a sinistra dell'immagine (in pixel).

L'attributo **tabindex** definisce l'ordine in cui i differenti field devono attivarsi quando il visitatore clicca sul tasto tab.

U N ESEMPIO:

Guarda questo esempio HTML:

```
<html>
<head>
<title>My Page</title>
</head>
<body>
<form name="myform" action="http://www.mydomain.com/myformhandler.cgi" method="POST">
<div align="center">
<br><br>
<input type="text" size="25" value="Scrivi qui il tuo nome!">
<br> <input type="image" src="go.gif" name="image" width="60" height="60"> <br>
</div>
</form>
</body>
</html>
```



E d ecco il risultato:

Inizio modulo



TABELLA RIASSUNTIVA

| HTML | SPIEGAZIONE | ESEMPIO |
|--|--|---------|
| textarea rows= cols= name= | Text area - diverse righe Righe nel field. Colonne nel field. Nome del field. | |
| wrap= off virtual physical | Controllo fine riga. Ignora gli "a capo". Inserisce gli "a capo", ma invia il testo come è stato inserito. Inserisce gli "a capo" quando è necessario e li invia. | |
| text size= maxlength= name= value= | Text field di una riga Caratteri mostrati. Max caratteri permessi. Nome del field. Valore iniziale nel field. | |
| password size= maxlength= name= value= | Password field. Caratteri mostrati. Caratteri autorizzati a entrare. Nome del field. Valore iniziale nel field. | |
| checkbox name= value= | Scegli una o più opzioni Nome del field. Valore iniziale nel field. | |
| radio name= value= | Scegli una sola opzione Nome del field. Valore iniziale nel field. | |
| select name= size= multiple= | Menù a discesa Nome del field. Numero di voci nella lista. Se sì, permette scelte multiple. | |
| option selected value= | Voci individuali nel menù. Creare una voce default. Valore da inviare se selezionato.. | |
| hidden name= value= | Non mostra il form. Nome del field. Valore da inviare. | |
| reset name= value= | Bottone per resettare tutti i field. Nome del bottone. Testo mostrato sul bottone. | |

| | | |
|---------------|--|---|
| submit | Bottone per inviare il form | |
| name= | Nome del bottone. |  |
| value= | Testo mostrato sul bottone. | |
| image | Immagine funzionante come bottone | |
| name= | Nome dell'immagine. |  |