

## **UNITÀ DIDATTICA**

### **LAVORIAMO CON HTML**

#### **Obiettivi**

In tale unità impareremo a conoscere il linguaggio HTML, conosceremo la struttura fondamentale di un documento e gli elementi fondamentali per scrivere una semplice pagina web.

#### **Sommario**

- 1 Cosa serve**
- 2 Struttura base di una pagina**
- 3 Formattare un documento**
- 4 Allineamento del testo**
- 5 Colore, font e grandezza dei caratteri**
- 6 Sfondi**

#### **Parole Chiave**

Tag, struttura, formattare, font, size, color, sfondo

---

## LAVORIAMO CON HTML

### 1.1. Cosa serve

Per scrivere un documento in HTML occorre:

- un computer
- un browser web
- un editor

Quando scriviamo una pagina web possiamo usare due approcci:

- scrivere il testo e i comandi di HTML con un semplice editor di testo
- scrivere il testo con un programma autore<sup>1</sup> che genera i comandi HTML in automatico.

Il primo modo permette di gestire meglio il codice, mentre usando un editor grafico si può incorrere nel problema di avere un “codice sporco”, ossia con istruzioni inserite in automatico e soprattutto di visualizzare le pagine in modo non perfettamente corretto.

Noi, inizialmente useremo un editor testuale per scrivere le prime pagine in HTML, successivamente realizzeremo il sito web usando uno specifico software.

Il computer e il browser web dovremmo necessariamente averli, mentre come editor testuale per scrivere la nostra prima pagina web useremo il “blocco note” o “Notepad” di Windows che troveremo in:

***Start → programmi → accessori → blocco note***

---

<sup>1</sup> Un programma autore è un software del tipo WYSIWYG (what you see is what you get – ciò che vedi è ciò che fai).

## 1.2. Struttura base di una pagina

Ogni pagina scritta usando HTML è un file la cui estensione è: nome\_file.html oppure nome\_file.htm.

Fino a un po' di tempo fa si attribuiva ai file l'estensione **htm**, ma questo avveniva perché il vecchio sistema operativo DOS e WINDOWS 3.1 non erano in grado di gestire i file la cui estensione superasse i 3 caratteri.

Pertanto nome\_file.html era diventato nome\_file.htm, così come anche altre estensioni avevano subito delle contrazioni (ad esempio nome\_immagine.jpeg era diventato nome\_immagine.jpg).

Il problema delle estensioni è stato superato con le versioni successive di Windows e di conseguenza oggi si può decidere equivalentemente se attribuire ai files estensione .HTML o .htm.

Se il file ha estensione \*.html o .htm, il browser dovrebbe essere in grado di aprirlo in automatico cliccandoci su due volte con il tasto sinistro del mouse.

Per modificare la pagina è possibile usare i comandi **Visualizza > HTML** utilizzando i menu del browser, cambiare il codice, salvare e utilizzare il pulsante "aggiorna" del browser per vedere le modifiche oppure aprire il programma "Blocco note" e dal menu **File -> Apri->** scegliere il file da modificare.

All'interno troveremo delle parole che iniziano con "<" e finiscono con ">" che rappresentano dei comandi ben precisi.

Il termine racchiuso tra i segni "<" e ">" si chiama TAG ed ha lo scopo di "etichettare" parti di testo in modo tale che il browser lo possa visualizzare in modo appropriato.

Il termine "< TAG" rappresenta il tag di apertura, ossia indica il punto in cui inizia il testo sul quale applicare un certo effetto, mentre "/TAG>" rappresenta il tag di chiusura, ossia rappresenta il punto in cui finisce il particolare effetto indicato.

Ci sono delle istruzioni fondamentali che bisogna necessariamente scrivere prima di qualsiasi altro testo.

Un documento HTML ha una la seguente struttura:

<code>&lt;HTML&gt;</code>	inizia una pagina web
<code>&lt;HEAD&gt;</code>	possono essere inserite delle meta istruzioni
<code>&lt;TITLE&gt;</code>	titolo della pagina che è visualizzato sulla barra del titolo
.....	
<code>&lt;/TITLE&gt;</code>	tag di chiusura
<code>&lt;/HEAD&gt;</code>	tag di chiusura
<code>&lt;BODY&gt;</code>	contiene tutti gli elementi di una pagina web
.....	
<code>&lt;/BODY&gt;</code>	tag di chiusura
<code>&lt;/HTML&gt;</code>	fine della pagina web

L'HTML è "case insensitive", cioè indipendente dal formato.

Questo significa che è del tutto indifferente scrivere i tag in maiuscolo o in minuscolo, pertanto `<title>` o `<TITLE>` è esattamente la stessa cosa.

Poiché, come abbiamo già detto in precedenza, l'evoluzione dell'HTML è l'XHTML che invece è "case sensitive", ossia sensibile al formato è consigliabile scrivere tutto in minuscolo, per abituarsi già al linguaggio che verrà.

Maiuscolo e minuscolo, in ogni caso non costituiscono errore.

In una pagina HTML, i tag possono essere annidati, pertanto è buona norma, per avere una buona leggibilità del codice usare l'indentazione (rientri a blocchi).

---

Altro suggerimento, utile quando più persone lavorano ad uno stesso progetto, è inserire dei commenti nei punti più significativi per rendere il codice leggibile

Un commento ha la seguente sintassi:

```
<!-- commento -->
```

ed è invisibile al browser, nel senso che pur essendo presente nel codice, non viene visualizzato sul monitor.

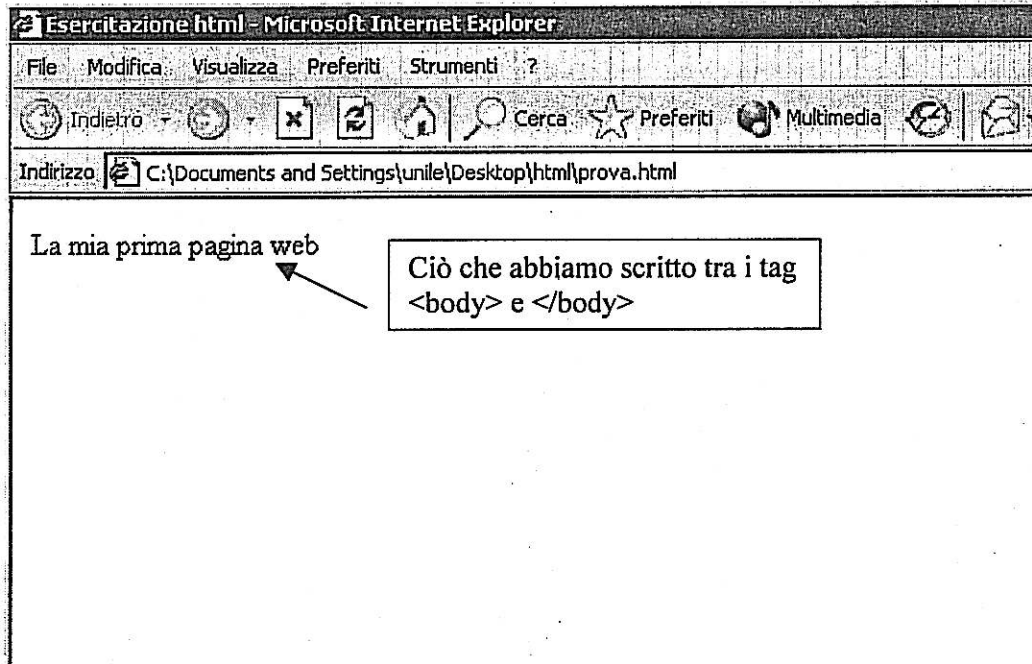
### 1.3. Formattare un documento

Cominciamo ora a scrivere una prima pagina con HTML, conferendo ad essa alcuni elementi di formattazione.

```
<HTML>
  <head>
    <title>
      Esercitazione HTML
    </title>
  </head>
  <body>
    La mia prima pagina web
  </body>
</HTML>
```

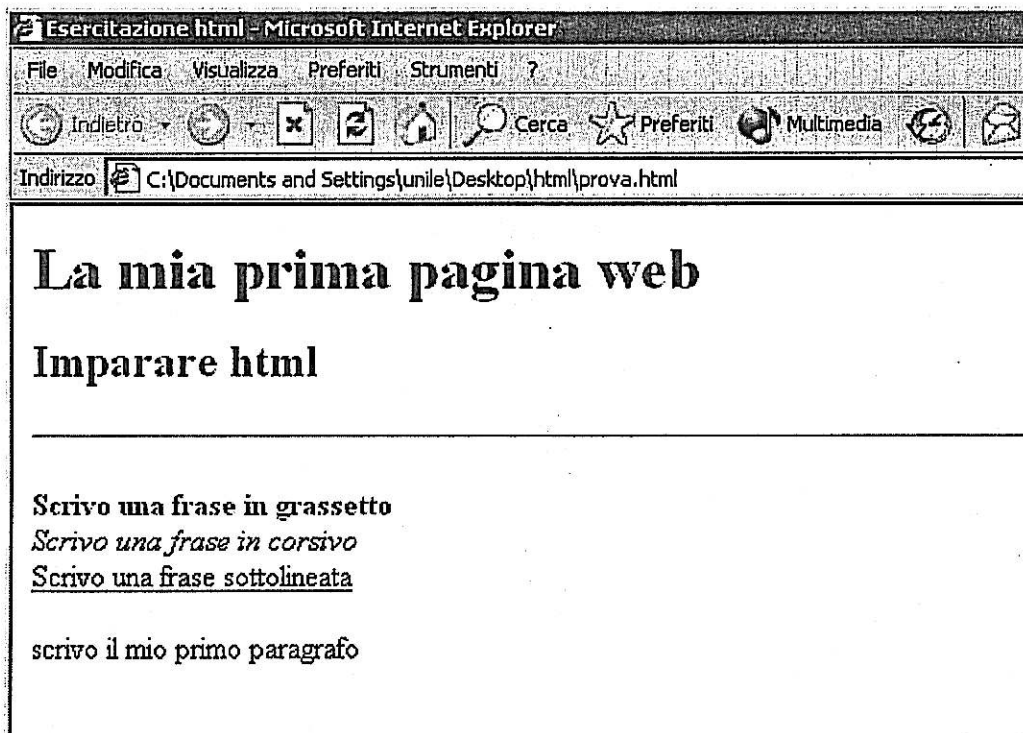
Salviamo il file chiamandolo: *prova.html* e apriamo il browser per visualizzarlo. Pertanto vedremo la seguente pagina web:

↙ Ciò che abbiamo scritto tra i tag <title> e </title>



E' una pagina che non presenta nessun elemento di rilievo.

Se vogliamo inserire elementi di formattazione per conferire alla pagina un aspetto più gradevole, allora avremo:



il cui corrispondente codice è:

```
<HTML>
  <head>
    <title>
      Esercitazione HTML
    </title>
  </head>
  <body>
    <h1> La mia prima pagina web </h1>
    <h2> Imparare HTML </h2>
    <hr>
    <b>Scrivo una frase in grassetto </b>
    <br>
    <i>Scrivo una frase in corsivo </i>
    <br>
    <u>Scrivo una frase sottolineata </u>
    <p>
    scrivo il mio primo paragrafo
    </p>
  </body>
</HTML>
```

Analizziamo in dettaglio quali sono i tag utilizzati:

TAG	Significato
<code>&lt;h1&gt;... &lt;/h1&gt;</code>	<b>h</b> sta per heading e serve per le intestazioni I tag di intestazione sono in ordine decrescente (dalla dimensione del testo più grande – h1 – alla dimensione più piccola – h6)
<code>&lt;hr&gt;</code>	Permette di inserire una linea orizzontale – Da notare che questo tag non ne prevede uno di chiusura
<code>&lt;br&gt;</code>	Permette di inserire una interruzione di riga - Da notare che questo tag non ne prevede uno di chiusura
<code>&lt;p&gt; ... &lt;/p&gt;</code>	Permette di passare ad un paragrafo diverso. Quando si passa da un paragrafo ad un altro viene saltata automaticamente una riga
<code>&lt;b&gt;... &lt;/b&gt;</code>	Permette di ottenere il grassetto
<code>&lt;i&gt; ... &lt;/i&gt;</code>	Permette di ottenere il corsivo
<code>&lt;u&gt; ... &lt;/u&gt;</code>	Permette di ottenere il sottolineato

#### 1.4. Allineamento del testo

Per poter disporre il testo a destra, al centro o a sinistra oltre al tag occorre specificare anche gli attributi.

Gli attributi sono sempre termini di codice utilizzati all'interno di alcuni tag e permettono di controllare meglio alcune opzioni.

Se ad esempio in un documento vogliamo disporre un paragrafo al centro, oppure un titolo a destra dovremmo usare attributi del tipo:

`<p align="center"> ..... </p>` per centrare un paragrafo

`<p align="right"> ..... </p>` per allineare a destra un paragrafo

`<p align="justify"> ..... </p>` per giustificare un paragrafo

se invece vogliamo disporre al centro un insieme di oggetti, allora useremo il tag `<DIV>` con la seguente sintassi



```
<div align="center">
```

```
<b> La mia pagina web </b>
```

```
<p> nuovo paragrafo </p>
```

```
</div>
```

Vediamo qual è la differenza tra i tag <P>, <DIV>:

**<P> lascia una riga prima e dopo la propria chiusura**

Ad esempio se scriviamo

```
<p>paragrafo 1</p>
```

```
<p>paragrafo 2</p>
```

quello che verrà visualizzato è:

paragrafo 1

paragrafo 2

**<DIV> non lascia righe prima e dopo la propria chiusura, ma - essendo un elemento di blocco - va a capo**

Se ad esempio scriviamo

```
<div>blocco 1</div>
```

```
<div>blocco 2</div>
```

quello che verrà visualizzato è:

blocco 1

blocco 2

### 1.5. Colore, font e grandezza dei caratteri

Se si vuole avere un controllo maggiore sul testo inserito relativo al tipo di font, alla dimensione e al colore si usano dei tag specifici.

Il tipo di carattere (cioè il "font") che il browser visualizza di default è il "Times New Roman" che non è molto indicato per il web perché ha non ha una buona leggibilità a causa degli effetti di rifinitura dei caratteri.

Si preferisce usare caratteri "senza grazie" (sans serif) ossia senza "quei particolari riccioli delle lettere come ad esempio l'Arial, il Verdana o l'Helvetica."<sup>2</sup>

Per scegliere il font con cui un carattere deve essere visualizzato la sintassi è:

```
<font face="Arial">testo in Arial</font>
```

Mentre per indicare alcune famiglie "sicure" di caratteri:

```
<font face="Arial, Helvetica, sans-serif">Arial e caratteri simili</font>
```

```
<font face="Times New Roman, Times, serif">Times e caratteri simili</font>
```

Se dobbiamo dichiarare oltre al font anche la grandezza e il colore la sintassi da usare è:

```
< font face="Arial, Helvetica, sans-serif" size="6" color ="red"> nostro testo </font>
```

<sup>2</sup> E' da sottolineare che non è corretto usare dei caratteri particolari, perché l'utente web potrebbe non averli installati sul suo computer.

Quando incontra dei caratteri non standard, il browser li interpreta facendo riferimento a quelli che somigliano di più scelti tra quelli presenti sul computer dell'utente e ciò può creare effetti indesiderati durante la visualizzazione della pagina.

E' buona norma, quando si sceglie un carattere:

- scegliere caratteri standard che sicuramente sono installati sul computer degli utenti
- non indicare un solo carattere, ma una famiglia di caratteri in modo che il browser cercherà di trovare nella propria cartella dei fonts il primo carattere indicato, se non lo trova passerà al secondo ecc.

L'attributo **size** può assumere un valore da 1 (piccolo) a 7 (grande)

In genere il valore di default visualizzato dai browser è 3.

Però usando i segni "+" e "-" è possibile aumentare o diminuire la grandezza rispetto al valore di default.

Ad esempio `<font size="+2">` significa che il testo verrà visualizzato due volte in più rispetto al valore standard.

L'attributo **color** può assumere sia il nome del colore (red, green, blue ecc) sia il corrispondente valore esadecimale<sup>3</sup> (FF00AA, AA66BB ecc.)

Ecco una tabella con la notazione esadecimale e testuale di alcuni colori:

colore	parola chiave	notazione esadecimale
arancione	orange	#FFA500
blu	blue	#0000FF
bianco	white	#FFFFFF
giallo	yellow	#FFFF00
grigio	gray	#808080
marrone	brown	#A52A2A
nero	black	#000000
rosso	red	#FF0000
verde	green	#008000
viola	violet	#EE82EE

<sup>3</sup> E' possibile definire colori personalizzati usando il formato RGB (R sta per red, G sta per green, B sta per blu) inserendo il corrispondente valore esadecimale.

Il formato è #rrggbb dove rr, gg,bb sono valori esadecimali composti da due cifre (da 00 a ff)relativi ai componenti di rosso, verde e blu.

E' da tenere presente che:

ff significa piena luminosità

cc significa luminosità all'80%

99 significa luminosità all'60%

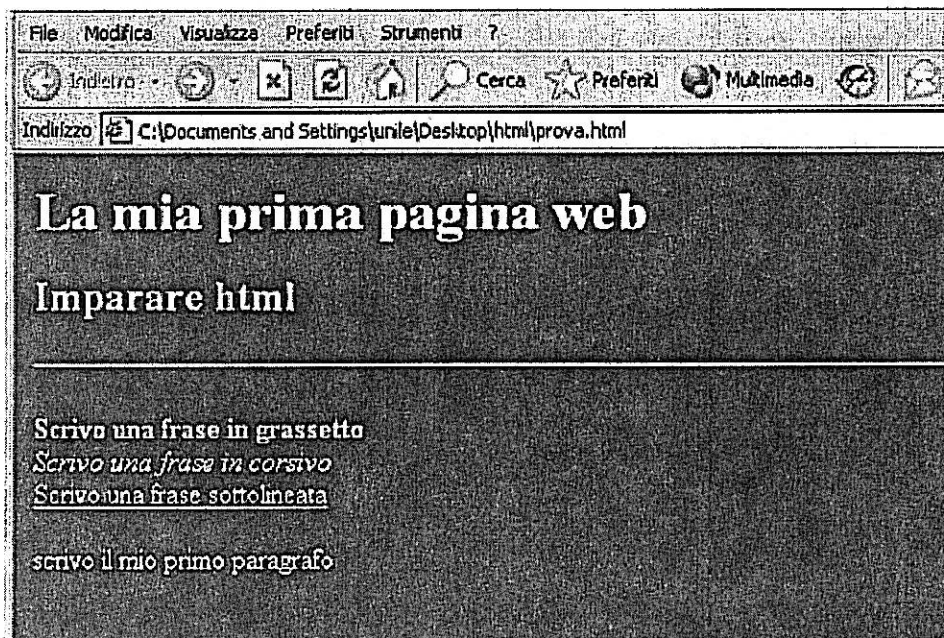
....

00 significa totale assenza del colore

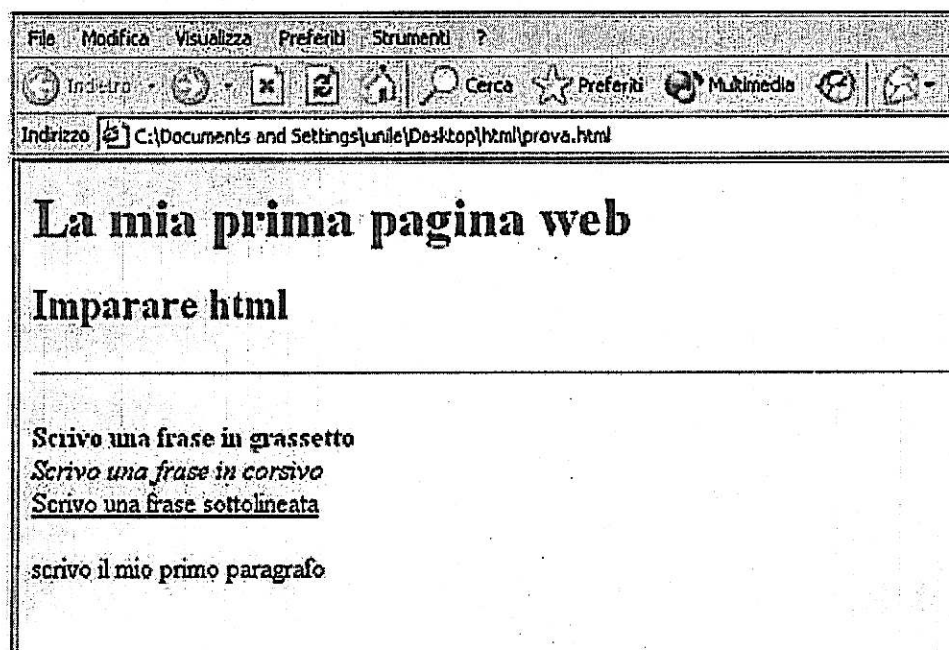
## 1.6. Sfondi

Gli esempi che abbiamo visto finora hanno avuto effetto solo sul testo mentre lo sfondo della pagina rimane bianco.

Vedremo ora come personalizzare lo sfondo di una pagina con un colore.



o con uno sfondo



---

Per avere un colore come sfondo il codice corrispondente è:

```
<body bgcolor="blue" text="white">
```

dove bgcolor sta per "background color" ossia "colore di sfondo".

Tuttavia non è consigliabile inserire la notazione del colore facendo riferimento a questo tipo di sintassi, dal momento che non possiamo sapere esattamente a quale tonalità di colore corrisponda il blu del computer dell'utente (ciò dipende dalle impostazioni locali del computer).

È preferibile in molti casi utilizzare la corrispondente codifica esadecimale del colore, che dà la possibilità di scegliere anche tonalità di colore non standard.

Pertanto con la notazione esadecimale si ha:

```
<body bgcolor="#0000FF" text="white">
```

Nell'esempio abbiamo anche impostato che il colore del testo, su uno sfondo blu, deve essere bianco usando l'attributo `text="white"`

Se si desidera attribuire a parti di testo, pur avendo impostato con l'istruzione precedente un colore di default, un colore diverso è possibile farlo usando il tag visto in precedenza

```
<font color=".....">
```

 e ricordandosi poi di chiudere il tag con `</font>`

Se invece si desidera inserire una immagine come sfondo il corrispondente codice è:

```
<body background="nome_immagine.jpg">
```

dove `nome_immagine` è il nome dell'immagine che deve essere inserita.

---

L'immagine può essere in formato IPEG oppure in formato GIF (dei formati delle immagini parremo successivamente).

Va sottolineato che l'immagine usata per lo sfondo deve essere in genere una immagine ben progettata, ossia non deve essere molto pesante (in termini di byte), non deve essere eccessiva, rischierebbe di distogliere l'attenzione dal contenuto o di stancare l'utente.

È anche possibile combinare i due attributi, in modo che mentre l'immagine di sfondo viene caricata, venga comunque visualizzata una colorazione della pagina:

```
<body bgcolor="#0000ff" background="nome_immagine.jpg">
```