

I fogli di stile

CSS

# Css

CSS è l'acronimo di *Cascading Style Sheets*

– *Fogli di stile in cascata*

Un foglio di stile specifica come un browser deve posizionare, formattare e visualizzare i vari elementi che compongono una pagina web

Un foglio di stile è una collezione di regole stilistiche che definiscono il look & feel (*apparenza visiva*, il *look*, *modalità di interazione* il *feel*) degli elementi.

– Uno stile può essere *interno* ad un file .html oppure può essere un *file di testo con estensione .css*)

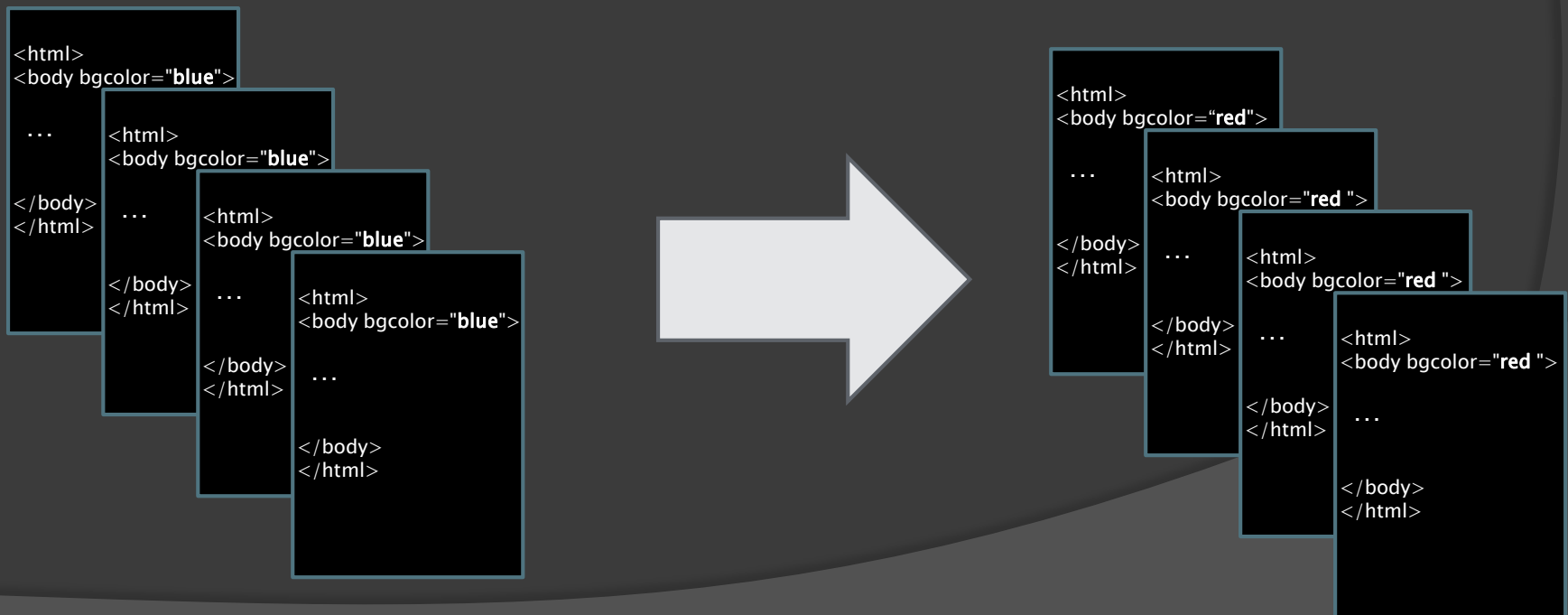
# CSS

Il browser fonde insieme le varie definizioni incontrate esaminando in cascata le definizioni incontrate (ecco il significato di *cascading*)

- Il termine *cascading* si riferisce all'ordine di applicazioni delle definizioni di stile
- Hanno *priorità minore* quelle in un file esterno, *maggiore* quelle all'interno dei tag

# CSS

Se ho costruito un sito con **50 pagine** e decido di **cambiare lo sfondo** di ogni pagina, quante pagine HTML devo aprire per **modificare il tag** relativo al colore dello sfondo?



# CSS

In passato

Le proprietà stilistiche dei vari elementi venivano suggerite al browser attraverso gli attributi dei tag

```
<BODY BGCOLOR="darkred">
```

In questo modo le pagine web sono confuse e difficili da leggere ed ognuna diversa da un'altra

# CSS

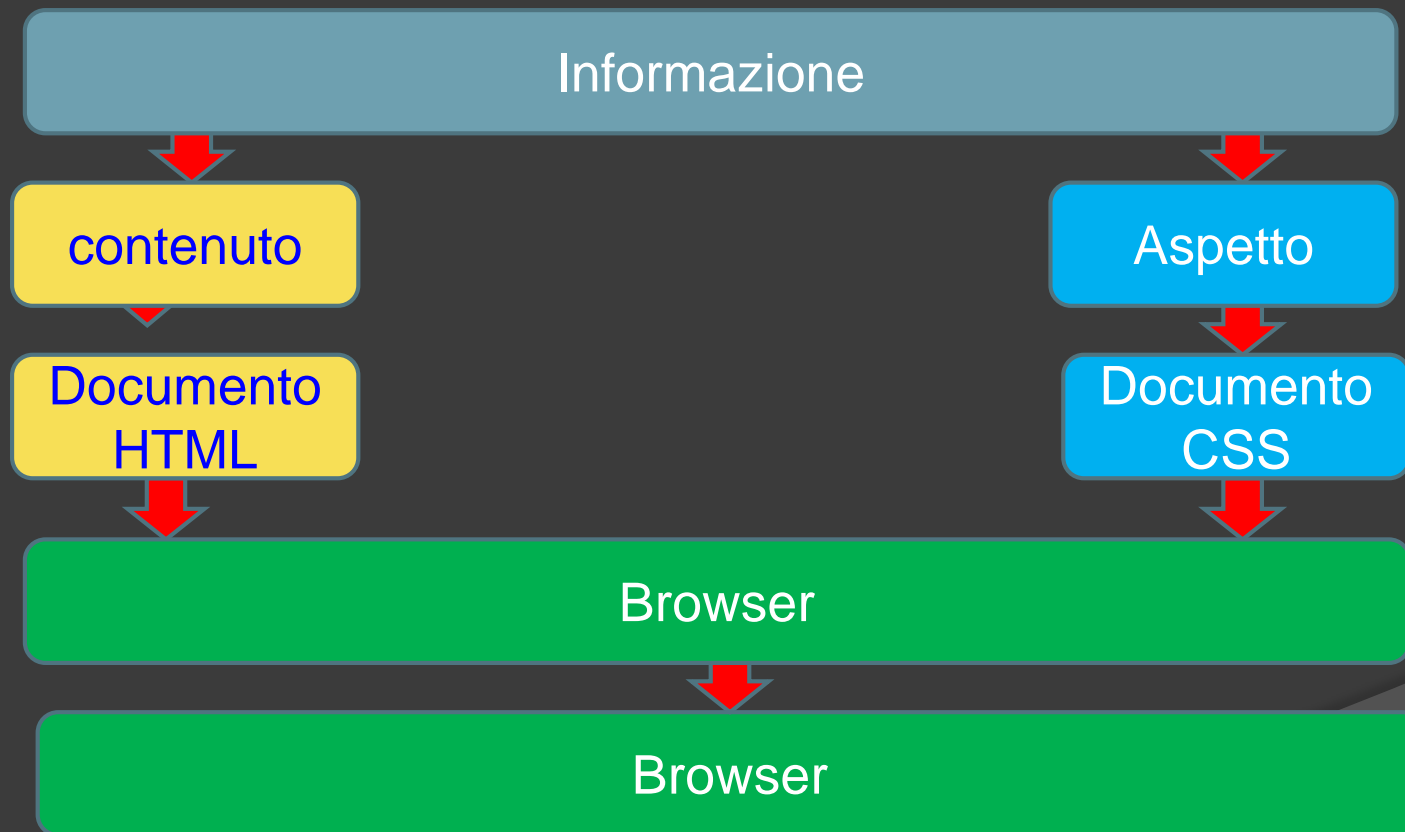
CSS permette agli autori di siti WEB di separare l'aspetto di pagine web dal loro contenuto.

Nel seguito impareremo

- come farlo
- perché è bene farlo
- perché non è bene continuare ad usare il vecchio modo per personalizzare pagine

# CSS

Separare il contenuto dell'informazione dal suo aspetto



# CSS

## Specifica dello stile

- Ogni elemento possiede un insieme di proprietà che caratterizzano la sua presentazione (design)
- La definizione dello stile consiste nell'assegnare valori a un insieme di proprietà



# CSS

Lo stile definisce le regole che guidano il browser nella visualizzazione del contenuto di un tag (suggerisco come mostrare il contenuto)

Con la netta separazione contenuto aspetto si guadagna:

- Chiarezza semantica
- Chiarezza stilistica
- Uno strumento potente e generale per specificare lo stile

# CSS

Ogni documento HTML può essere visto come un insieme di blocchi (contenitori) sui quali si può agire con stili diversi

I tag HTML definiscono i blocchi

# CSS

```
<HTML>
```

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

```
<BODY>
```

```
<H1>titolo</H1>
```

```
<P> primo paragrafo </P>
```

```
<P> secondo paragrafo </P>
```

```
<P> terzo paragrafo <A HREF="pippo.html"> link</A></P>
```

```
</BODY>
```

```
</HTML>
```

# CSS

## Come specificare uno stile?

Esistono tre modi per specificare uno stile

– All'interno di elementi HTML

- **Inline Style**

– All'interno del tag <head>

- **Internal Style Sheet**

– In un file esterno

- **External Style Sheet**

# CSS

## Stile in linea – Inline Style

HTML permette di assegnare gli stili agli elementi in tre modi:

- Tutti gli elementi con lo stesso nome hanno lo stesso stile
- Tutti gli elementi di una certa categoria hanno lo stesso stile (si usa l'attributo CLASS)
- Uno specifico elemento ha un dato stile (l'elemento è identificato dal valore dell'attributo ID)

# CSS

## Stile in linea – Inline Style

Lo stile *inline* lo si definisce all'interno di un qualsiasi tag HTML

– Permette di definire proprietà di un elemento specifico

```
<P STYLE="font-size:120%; text-align:center; background:red;" >
```

Assegnando ad ogni proprietà un valore

## ESEMPIO

Lo stile *inline* indica la formattazione di un singolo elemento

La definizione vale solo per il contenuto del tag

```
<H1 STYLE="color:blue;">
```

```
la scritta blue</H1>
```

```
<H1>la scritta normale</H1>
```



## Internal Style Sheet

lo stile è *incorporato nel documento HTML*. Le definizioni vanno all'interno del tag contenitore

```
<STYLE> ... </STYLE>
```

che a sua volta deve essere contenuto dal tag

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

La definizione si applica a tutti gli elementi della pagina

```
<STYLE TYPE="text/css">
```

```
P { font-size:120%; text-align:center; background:red;}
```

```
</STYLE>
```



```
<HTML>
<HEAD>
<TITLE>Esempio</TITLE>
<STYLE type="text/css">
P { font-size:120% ;text-align:center;
background:red;}
</STYLE>
</HEAD>
<BODY>
<P> Questo è un esempio di stile internal </P>
</BODY>
</HTML>
```



# I fogli di stile Internal Style Sheet

Ogni foglio di stile consiste di una serie di istruzioni chiamate **statements**.

**Uno statement** fa due cose:

- Identifica gli **elementi (P, BODY, UL, ...)** in un documento HTML che lo includono
- Suggerisce al browser come “mostrare” questi elementi

Lo schema è il seguente:

Selector ( property: value;)

Mediante l'attributo style, nella configurazione in-line si fornisce al selettore body il valore rosso alla proprietà background-color

```
<body style="background-color: #FF0000;">
```

# Css: proprietà e valori

## Colori e sfondi

[color](#)

[background-color](#)

[background-image](#)

[background-repeat](#)

[background-attachment](#)

[background-position](#)

[background](#)

La proprietà **color** descrive il colore di primo piano di un elemento. Per esempio, immagina di volere tutti i titoli del documento di colore rosso scuro.

I titoli sono tutti marcati con l'elemento HTML `<h1>`.

Il codice sotto definisce il colore rosso dell'elemento `<h1>`.

```
h1 { color: #ff0000; }
```

- [Visualizza l'esempio](#)

I colori possono essere inseriti in valore esadecimale come nell'esempio sopra, o puoi usare il nome del colore in inglese ("red") o il suo valore in rgb (rgb(255,0,0))

## La proprietà 'background-color'

La proprietà `background-color` descrive il colore di sfondo degli elementi.

L'elemento `<body>` contiene tutto il contenuto di un documento HTML. Per questo motivo, per cambiare il colore di sfondo di tutta la pagina, dovremmo applicare la proprietà `background-color` all'elemento `<body>`.

Il colore di sfondo può essere applicato anche ad altri elementi compresi i titoli e il testo. Nell'esempio sotto sono stati applicati diversi colori di sfondo agli elementi `<body>` e `<h1>`.

```
body { background-color: #FFCC66; }
```

```
h1 { color: #990000; background-color: #FC9804; }
```

- [Visualizza l'esempio](#)

Nota che abbiamo applicato due proprietà ad `<h1>` separandole con un punto e virgola.

## Immagini di sfondo [**background-image**]

La proprietà CSS **background-image** viene usata per inserire una immagine di sfondo

Per inserire l'immagine della farfalla come immagine di sfondo di una pagina web applica semplicemente la proprietà `background-image` al `<body>` e specifica la posizione dell'immagine.

```
body { background-color: #FFCC66; background-image: url("butterfly.gif"); }  
h1 { color: #990000; background-color: #FC9804; }
```

• [Visualizza l'esempio](#)

NB: Nota come abbiamo specificato l'indirizzo dell'immagine come `url("butterfly.gif")`. Questo significa che l'immagine è posizionata nella stessa cartella del foglio di stile. Puoi riferirti anche ad immagini in altr cartelle usando `url("../images/butterfly.gif")` o anche su Internet indicando l'indirizzo completo del file: `url("http://www.html.net/butterfly.gif")`.

## Immagini di sfondo ripetute [background-repeat]

Nell'esempio sopra hai notato che per default la farfalla viene ripetuta sia orizzontalmente che verticalmente in modo da coprire tutto lo schermo? La proprietà background-repeat controlla questo comportamento.

La tabella sotto sottolinea i quattro diversi valori che può assumere background-repeat. Per esempio, per evitare ripetizione di un'immagine di sfondo il codice dovrebbe apparire come questo:

```
body { background-color: #FFCC66; background-image: url("butterfly.gif"); background-repeat: no-repeat; } h1 { color: #990000; background-color: #FC9804; }
```

### •[Visualizza l'esempio](#)

Valore	Descrizione	Esempio
background-repeat: repeat-x	L'immagine viene ripetuta orizzontalmente	<a href="#">Visualizza l'esempio</a>
background-repeat: repeat-y	L'immagine viene ripetuta verticalmente	<a href="#">Visualizza l'esempio</a>
background-repeat: repeat	L'immagine viene ripetuta sia orizzontalmente che verticalmente	<a href="#">Visualizza l'esempio</a>
background-repeat: no-repeat	L'immagine non viene ripetuta	<a href="#">Visualizza l'esempio</a>



## Immagini di sfondo bloccate [background-attachment]

La proprietà background-attachment specifica se una figura di sfondo è fissa o si muove insieme all'elemento che la contiene.

Una immagine di sfondo fissa non si muoverà con il testo quando il lettore scorrerà la pagina, mentre una immagine di sfondo non bloccata si muoverà insieme al testo della pagina web.

La tabella sotto sottolinea i due valori che può avere background-attachment.

Fai click sugli esempi per vedere le differenze fra scroll e fixed.

Il codice sotto, per esempio, blocca l'immagine di sfondo.

```
body { background-color: #FFCC66; background-image: url("butterfly.gif");  
background-repeat: no-repeat; background-attachment: fixed; } h1 { color:  
#990000; background-color: #FC9804; }
```

•[Visualizza l'esempio](#)

Valore	Descrizione	Esempio
Background-attachment: scroll	L'immagine si muove con la pagina - non bloccata	<a href="#">Visualizza l'esempio</a>
Background-attachment: fixed	L'immagine è bloccata	<a href="#">Visualizza l'esempio</a>

## Immagini di sfondo posizionate [background-position]

Per default una immagine di sfondo verrà posizionata nell'angolo in alto a sinistra dello schermo. La proprietà background-position ti permette di cambiare il default e posizionare l'immagine di sfondo in qualsiasi altro posto dello schermo che preferisci.

Ci sono molti modi per stabilire i valori di background-position. Tutti comunque stabiliscono un insieme di coordinate. Per esempio, il valore '100px 200px' posiziona l'immagine a 100px dal lato sinistro e a 200px dal bordo superiore della finestra del browser.

Le coordinate possono essere indicate come percentuale delle dimensioni dello schermo, unità fisse (pixel, centimetri, ecc.) o puoi usare le parole inglesi top, bottom, center, left e right. Il modello sotto illustra meglio il sistema:

La tabella ti da qualche esempio.

Il codice di esempio sotto posiziona l'immagine di sfondo nell'angolo in basso a destra:

```
body { background-color: #FFCC66; background-image: url("butterfly.gif");  
background-repeat: no-repeat; background-attachment: fixed; background-  
position: right bottom; } h1 { color: #990000; background-color: #FC9804; }
```

• [Visualizza l'esempio](#)

Valore	Descrizione	Esempio
background-position: 2cm 2cm	L'immagine è posizionata a 2 cm dalla sinistra e a 2 cm dal basso della pagina	<a href="#">Visualizza l'esempio</a>
background-position: 50% 25%	L'immagine è centrata ad un quarto della pagina dal basso	<a href="#">Visualizza l'esempio</a>
background-position: top right	L'immagine è posizionata nell'angolo in alto a destra della pagina	<a href="#">Visualizza l'esempio</a>

## Raggruppamento **[background]**

La proprietà **background** è una scorciatoia per tutte le proprietà di sfondo che abbiamo elencato in questa lezione.

Con la proprietà **background** puoi raggruppare molte proprietà insieme e scrivere così il tuo foglio di stile in una maniera ancora più corta e più facile da leggere.

Guarda per esempio queste cinque linee:

```
background-color: #FFCC66;  
background-image: url("butterfly.gif");  
background-repeat: no-repeat;  
background-attachment: fixed;  
background-position: right bottom;
```

Usando **background** puoi ottenere lo stesso risultato con solo una linea di codice:

```
background: #FFCC66 url("butterfly.gif") no-repeat fixed right bottom;
```

Di seguito l'ordine con cui devono essere inserite le proprietà:

```
[background-color] | [background-image] | [background-repeat] | [background-attachment] | [background-position]
```

Se una proprietà non viene definita verrà automaticamente posta al suo valore di default. Per esempio, **background-attachment** e **background-position** non vengono definite nell'esempio:

```
background: #FFCC66 url("butterfly.gif") no-repeat;
```

Queste due proprietà, che non sono state specificate, verranno soltanto definite con il loro valore di default che, come abbiamo visto, è **scroll** e **top left**

## Font

Verranno descritte le seguenti proprietà CSS:

[font-family](#)

[font-style](#)

[font-variant](#)

[font-weight](#)

[font-size](#)

[font](#)

## La famiglia dei caratteri [font-family]

La proprietà font-family viene usata per definire le priorità per la lista dei caratteri da usare per visualizzare un elemento o una pagina web. Se il primo font della lista non è stato installato sul computer che accede al sito, allora verrà provato il secondo font della lista e così via finché non verrà trovato un fonte disponibile.

Ci sono due tipi di nomi per catalogare i font: i nomi della famiglia e le famiglie generiche. I due termini vengono spiegati sotto.

### Nome della famiglia

Esempi di nomi della famiglia (conosciuto spesso con il termine "font") possono essere per esempio "Arial", "Times New Roman" o "Tahoma".

### Famiglia generica

Le famiglie generiche possono essere descritte meglio come gruppi di nomi di famiglie con caratteristiche uniformi. Un esempio è il sans-serif (in italiano, senza grazie), che è la collezione dei font senza le cosiddette grazie, o "piedi".

la lista dei font deve iniziare con i font più usati e poi con alcuni font alternativi. E' raccomandabile terminare la lista con il nome di una famiglia generica di font.

In questo modo la pagina potrà essere visualizzata con un font della stessa famiglia generica, nel caso in cui tutti i font listati non fossero disponibili.

Un esempio di lista di priorità dei font potrebbe essere fatta così:

```
h1 {font-family: arial, verdana, sans-serif;} h2 {font-family: "Times New Roman", serif;}
```

• [Visualizza l'esempio](#)

I titoli marcati con <h1> verranno visualizzati con il font "Arial". Se questo font non è stato installato sul computer dell'utente, allora verrà utilizzato al suo posto il font "Verdana". Se entrambi questi font non sono disponibili, i titoli verranno visualizzati utilizzando un font della famiglia **sans-serif**.

Nota come il nome del font "Times New Roman" contiene degli spazi e per questo motivo viene messo tra doppi apici.

Times New Roman Garamond Georgia	Queste tre famiglie di font appartengono alla famiglia generica serif (con grazie). La loro caratteristica è quella di avere "piedi" (o grazie, in italiano).
Trebuchet Arial Verdana	Queste tre famiglie di font appartengono alla famiglia generica sans-serif (senza grazie). La loro caratteristica è quella di non avere "piedi" (o grazie, in italiano).
Courier Courier New Andale Mono	Queste tre famiglie di font appartengono alla famiglia generica monospace. La loro caratteristica è quella di avere una larghezza fissa.

## Stile del font [font-style]

La proprietà font-style definisce lo stile **normale**, **italico** o **obliquo** del font scelto. Nell'esempio sotto tutti i titoli marcati con <h2> verranno visualizzati in italico.

```
h1 {font-family: arial, verdana, sans-serif;} h2 {font-family: "Times New Roman", serif; font-style: italic;}
```

- [Visualizza l'esempio](#)



## Variante del font [font-variant]

La proprietà font-variant viene usata per scegliere tra le varianti **normal** o **small-caps** di un font. Un font **small-caps** è un font font che usa tutte lettere maiuscole di dimensioni leggermente più piccole, al posto delle lettere minuscole. Confuso? Guarda questo esempio:

Se viene assegnato a font-variant il valore **small-caps** e non è disponibile nessun font small-caps il browser molto probabilmente mostrerà il testo tutto maiuscolo.

```
h1 {font-variant: small-caps;} h2 {font-variant: normal;}
```

## Peso del font [font-weight]

La proprietà font-weight descrive il grado di neretto, o "di pesantezza", che dovrebbe avere il font. Un font può essere sia **normal** che **bold**. Alcuni browser supportano anche l'uso di una serie di numeri tra 100 e 900 (in centinaia) per descrivere la pesantezza di un font.

```
p {font-family: arial, verdana, sans-serif;} td {font-family: arial, verdana, sans-serif;  
font-weight: bold;}
```

- [Visualizza l'esempio](#)

## Dimensione del font [font-size]

La dimensione di un font viene definita con la proprietà font-size.

Per descrivere le dimensioni di un font si può scegliere tra diverse unità di misura (per esempio, pixel e percentuali). In questo tutorial ci focalizzeremo sulle unità più utilizzate comunemente e più appropriate, mostrate negli esempi:

```
h1 {font-size: 30px;} h2 {font-size: 12pt;} h3 {font-size: 120%;} p {font-size: 1em;}
```

### •[Visualizza l'esempio](#)

C'è una sola differenza chiave tra le quattro unità elencate sopra. Le unità 'px' e 'pt' indicano dimensioni assolute per il font, mentre '%' e 'em' permettono all'utente di ridimensionare il font come preferiscono. Molti utenti sono diversamente abili, anziani o semplicemente vedono poco o hanno un monitor di qualità scadente. **Per far sì che il web sia accessibile a tutti**, dovresti usare le unità ridimensionabili come '%' o 'em'.

## Raggruppamento [font]

Usando la proprietà font è possibile raggruppare tutte le diverse proprietà dei font in una sola.

Per esempio immagina queste quattro linee di codice usate per descrivere le proprietà del font per il tag <p>:

```
p { font-style: italic; font-weight: bold; font-size: 30px; font-family: arial, sans-serif; }
```

Usando la proprietà per il raggruppamento il codice sarà semplificato:

```
p { font: italic bold 30px arial, sans-serif; }
```

L'ordine dei valori per font è:

```
font-style | font-variant | font-weight | font-size | font-family
```

## Testo

Formattare e aggiungere dello stile al testo è una delle prerogative principali per ogni web designer. In questa lezione ti verranno introdotte alcune sorprendenti opportunità fornite dai CSS per aggiungere stile al testo. Descriveremo le seguenti proprietà:

[text-indent](#)

[text-align](#)

[text-decoration](#)

[letter-spacing](#)

[text-transform](#)

## Indentazione del testo [text-indent]

La proprietà text-indent ti permette di aggiungere un tocco elegante ai paragrafi di testo, applicando una indentazione alla prima linea del paragrafo.

Nell'esempio sotto abbiamo applicato una indentazione di **30px** a tutti i paragrafi marcati con il tag <p>:

```
p { text-indent: 30px; }
```

- [Visualizza l'esempio](#)

## Allineamento del testo [text-align]

La proprietà text-align corrisponde all'attributo align usato nelle precedenti versioni di HTML. Il testo può essere allineato a **sinistra**, a **destra** o **centrato**. Oltre a questo il valore **justify** permetterà di allineare ogni linea del testo sia a destra che a sinistra. Già conosci questo tipo di allineamenti da esempi sui giornali o sulle riviste.

Nell'esempio sotto il testo nel titolo della tabella <th> verrà allineato a destra mentre i dati nella tabella <td> saranno centrati. Inoltre i normali paragrafi di testo saranno giustificati:

```
th { text-align: right; } td { text-align: center; } p { text-align: justify; }
```

- [Visualizza l'esempio](#)

## Decorazione del testo [text-decoration]

La proprietà text-decoration permette di aggiungere al testo diverse "decorazioni" o "effetti". Per esempio puoi sottolineare il testo, inserire una linea sul testo o sopra il testo, ecc. Nell'esempio che segue i titoli <h1> vengono sottolineati con una linea sopra il testo e i titoli <h3> con una linea sul testo.

```
h1 { text-decoration: underline; } h2 { text-decoration: overline; } h3 {  
text-decoration: line-through; }
```

- [Visualizza l'esempio](#)

## Spazio fra i caratteri [letter-spacing]

Lo spazio tra i caratteri del testo può essere specificato usando la proprietà letter-spacing. Il valore della proprietà è semplicemente la distanza desiderata. Per esempio, se vuoi uno spazio tra i caratteri del testo di un paragrafo <p> di **3px** e una distanza tra i caratteri del titolo <h1> di **6px**, il codice che dovresti usare sarà:

```
h1 { letter-spacing: 6px; } p { letter-spacing: 3px; }
```

- [Visualizza l'esempio](#)



## Trasformazione del testo [text-transform]

La proprietà text-transform controlla l'uso delle maiuscole in un testo. Puoi scegliere **capitalize**, usare **uppercase** o **lowercase** a seconda di come appare il testo nel codice HTML.

Un esempio potrebbe essere la parola "headline" che può essere mostrata all'utente come "HEADLINE" o "Headline". Esistono quattro possibili valori per la proprietà text-transform:

**capitalize**

Usa la maiuscola per il primo carattere di ogni parola. Per esempio: "john doe" sarà "John Doe".

**uppercase**

Converte tutti i caratteri in maiuscole. Per esempio: "john doe" sarà "JOHN DOE".

**lowercase**

Converte tutte i caratteri in minuscole. Per esempio: "JOHN DOE" sarà "john doe".

**none**

Nessuna trasformazione - il testo viene presentato come appare nel codice HTML.

Come esempio useremo una lista di nomi. I nomi sono tutti marcati con <li> (list-item). Diciamo che vogliamo i nomi in lettere maiuscole e i titoli in minuscolo.

Cerca di guardare il codice HTML di questo esempio e vedrai che in effetti il testo è in lettere minuscole.

```
h1 { text-transform: uppercase; } li { text-transform: capitalize; }
```

• [Visualizza l'esempio](#)

## **Lezione 6: Link**

Puoi applicare tutto quello che hai imparato nelle lezioni precedenti anche ai link (cioè, cambiare il colore, i font, la sottolineatura, ecc.). La cosa nuova è che i CSS ti permettono di definire queste proprietà in modo diverso a seconda se il link è stato visitato o no, è attivo o se il cursore è proprio sopra il link. In questo modo è possibile aggiungere un po' di fantasia ed utili effetti al tuo sito web. Per controllare questi effetti devi usare quella che viene chiamata una pseudo-classe (pseudo-classes).

### **Cosa è una pseudo-classe?**

Una pseudo-classe ti permette di tenere in considerazione condizioni o eventi diversi quando definisci una proprietà per una tag HTML.

Guardiamo un esempio. Come sai, in HTML i link vengono specificati con il tag `<a>`. Possiamo quindi usare `a` come selettore nei CSS:

```
a { color: blue; }
```

Un link può assumere diversi stati. Per esempio, può essere già stato visitato o no. Puoi usare una pseudo classe per assegnare uno stile diverso ai link visitati e ai link non ancora visitati.

```
a:link { color: blue; } a:visited { color: red; }
```

Usa `a:link` e `a:visited` rispettivamente per i link non ancora visitati e per i link già visitati. I link attivi avranno come pseudo-classe `a:active`, mentre `a:hover` verrà usato per definire il cursore sopra il link.

Adesso vedremo in dettaglio ciascuna delle quattro pseudo-classi.

## Pseudo-classe: link

La pseudo-classe `:link` viene usata per i link che non sono stati ancora visitati dall'utente.

Nel codice di esempio riportato sotto i link non visitati appariranno di colore blu chiaro.

```
a:link { color: #6699CC; }
```

• [Visualizza l'esempio](#)

## Pseudo-classe: visited

La pseudo-classe `:visited` viene usata per i link che sono già stati visitati dall'utente.

Per esempio, nel codice sotto faremo tutti i link visitati con un colore viola scuro:

```
a:visited { color: #660099; }
```

• [Visualizza l'esempio](#)

## Pseudo-classe: active

La pseudo-classe `:active` viene usata per i link attivi.

Nell'esempio inseriamo un colore di sfondo giallo per i link attivi:

```
a:active { background-color: #FFFF00; }
```

• [Visualizza l'esempio](#)

## Pseudo-classe: hover

La pseudo-classe `:hover` viene usata quando il cursore del mouse passa sopra ad un link.

Questa proprietà può essere usata per creare un effetto interessante. Per esempio,

## Esempio 1: quando il cursore è sopra un link

E' molto diffuso creare effetti diversi quando il cursore passa sopra un link. Guarderemo quindi altri esempi legati alla pseudo-classe `:hover`.

### Esempio 1a: Spazio tra i caratteri

Come ricorderai dalla [lezione 5](#), lo spazio tra i caratteri può essere modificato usando la proprietà `letter-spacing`. E' possibile applicare questa proprietà ai link per creare degli effetti speciali:

```
a:hover { letter-spacing: 10px; font-weight:bold; color:red; }
```

• [Visualizza l'esempio](#)

### Esempio 1b: MAIUSCOLE e minuscole

Nella [lezione 5](#) abbiamo visto anche la proprietà `text-transform` che permette di rendere le lettere maiuscole o minuscole. Anche questa proprietà può essere usata per creare un effetto particolare sui link:

```
a:hover { text-transform: uppercase; font-weight:bold; color:blue; background-color:yellow; }
```

• [Visualizza l'esempio](#)

I due esempi sopra ti danno un'idea delle infinite possibilità che si hanno nel combinare insieme le diverse proprietà. Puoi creare i tuoi effetti speciali - prova!

## Esempio 2: Elimina la sottolineatura dai link

Una domanda molto frequente è come si fa per eliminare la sottolineatura dai link?

**Dovresti considerare attentamente se è necessario eliminare la sottolineatura dei link poichè questo potrebbe diminuire significativamente**

**l'usabilità del tuo sito web.** Le persone sono abituate a vedere i link sottolineati di blu e sanno che possono cliccarci sopra. Perfino mia madre lo sa! Se cambi la

In alternativa puoi definire la proprietà `text-decoration` insieme alle altre proprietà per tutte le quattro pseudo-classi.

```
a:link { color: blue; text-decoration:none; }
```

```
a:visited { color: purple; text-decoration:none; }
```

```
a:active { background-color: yellow; text-decoration:none; }
```

```
a:hover { color:red; text-decoration:none; }
```



La parte dello statement che identifica gli elementi è detto **selettore**. I **selettori** *selezionano gli elementi di una pagina.*

La parte dello statement che suggerisce al browser come gli elementi dovrebbero essere mostrati all'utente è detta **dichiarazione**

Una dichiarazione può contenere un gran numero di **proprietà**, **le parti** individuali di stile da applicare all'elemento selezionato.



## External Style Sheet

Sono definiti in un altro documento con estensione “.css”

Uno dei modi per importare uno stile esterno è usare il tag **<link>**

Il documento a cui si vuole applicare uno stile esterno deve contenere un collegamento (attraverso il tag <LINK>) al foglio di stile  
Il tag <LINK> deve essere usato all'interno di <HEAD>

Sintassi:

```
<link rel="stylesheet"  
type="text/css"  
href="NomeFileStile.css">
```

## Attributi di <LINK>

### rel

– Indica il tipo di file, se è un foglio di stile (stylesheet) oppure la definizione di un font da scaricare via rete (fontdef)

### type

– Specifica il tipo di foglio di stile CSS (text/css) oppure javascript (text/javascript)

### href

– Indica il nome ed il percorso che individuano il file stesso

## Sintassi dei fogli di stile – 1

1. Ogni istruzione (*statement*) deve avere un *selettore* ed una dichiarazione. La dichiarazione viene subito dopo il selettore ed è contenuta in una coppia di parentesi graffe
2. La dichiarazione consiste di una o più proprietà separate da punto e virgola (;)
3. Ogni proprietà ha un nome seguito da due punti (:) seguiti dal valore di quella proprietà. Ci sono vari tipi di valore, ma ogni data proprietà può assumerne solo alcuni come indicato dalla specifica CSS2.

# CSS

4. Se una proprietà può assumere più valori contemporaneamente, (ad esempio font-family) allora valori nella lista devono essere separati da virgola e da uno spazio
5. Se un valore assume un'unità di misura come valore attuale, (ad esempio font-size) allora non si deve mettere uno spazio tra il valore e l'unità di misura
6. Se un valore è costituito da più parole, allora circondare il valore con le virgolette
7. Come per HTML, si possono aggiungere spazi bianchi per rendere i fogli di stile più facili da leggere e da scrivere

# CSS

Per indicare un commento in un foglio di stile esterno si usa la stessa sintassi del C

- Inizio commento `/*`
- Fine commento `*/`

Per indicare un commento in un foglio di stile interno si usa la stessa sintassi dell'HTML

- Inizio commento `<!--`
- Fine commento `-->`

# CSS

Che cos'è un foglio css?

È un **documento di testo** con estensione **.css** in cui sono definite delle regole:

selettore

Proprietà valore



```
/* Definisco le regole per il corpo del documento */  
body {  
    background: White; /* Prima dichiarazione: notate il punto e virgola finale!*/  
    font-family: Verdana, Geneva, Arial, Helvetica, sans-serif; /* Seconda  
dichiarazione*/  
    font-size: 12px; /* Terza dichiarazione */  
}  
  
/* Titolo di primo livello */  
h1 {  
    color: black;  
}
```

# CSS

- Un **selettore** serve a **selezionare** la parte o le parti di un documento **soggette** ad una **specifica regola**. **Esso può essere:**
- Un tag predefinito (<H1>, <P>,<TD>)
- Un gruppo di elementi identificati da una specifica classe definita con l'attributo **Class**
- Un elemento definito univocamente con **ID**
- **Esempio**

```
h1 {color: #000000;}  
h2 {background: white;}  
h3 {background: white;}  
p {background: white; font: 12px Verdana, arial, sans-serif;}  
table {width: 200px;}
```





# CSS

<http://lau.csi.it/risorse/CSS2/index.shtml>

# Selettore

- ▶ Un **selettore** serve a **selezionare** la parte o le parti di un documento **soggette** ad una **specifica regola**. **Esso può essere:**
- ▶ Un tag predefinito (**<H1>**, **<P>**, **<TD>**)
- ▶ Un gruppo di elementi identificati da una specifica classe definita con l'attributo **Class**
- ▶ Un elemento definito univocamente con **ID**

## ▶ Esempio

### ◦ Sintassi

```
h1 {color: #000000;}  
h2 {background: white;}  
h3 {background: white;}  
p {background: white; font: 12px Verdana, arial, sans-serif;}  
table {width: 200px;}
```

# Due selettori speciali: Id e classi

- La **classe** è un selettore **definito dall'utente**:

```
.testorosso {  
font: 12px arial, Helvetica, sans-serif;  
color: #FF0000;  
}
```

- L'**Id** è usato per identificare in **modo univoco** un elemento:

```
#titolo {color: blue;}
```

- ⦿ Per ogni elemento si possono definire **almeno tre colori**:
  - **il colore di primo piano** (foreground color)
  - **il colore di sfondo** (background color)
  - **il colore del bordo** (border color)
- ⦿ Esempio:

```
p {color: black; background-color: white; }
```