CSS
(Cascading Style Sheets)

Antonio Lioy
< lioy@polito.it >

Politecnico di Torino
Dip. Automatica e Informatica

HTML (Hyper Text Markup Language)

- **linguaggio di descrizione della pagina**
- **descrizione testuale (file di testo)**
- **specifica tramite comandi (tag) interpretati dai browser**
- **i file HTML contengono tre tipi di informazioni:**
  - il testo vero e proprio del documento da visualizzare
  - una suddivisione logica del documento in parti (titolo, paragrafi, ...)
  - indicazioni sulla presentazione grafica del documento (formattazione, ovvero lo stile di visualizzazione)

HTML: stili

- **i tag furono originariamente pensati per definire il contenuto logico del documento e non la formattazione (che veniva lasciata al browser)**
  - `<h1>` vuol dire “questo è un titolo di primo livello”
  - `<p>` vuol dire “qui inizia un paragrafo”
  - e così via
- **sono stati aggiunti nuovi tag per la gestione della formattazione (ad es. `<font>`) indipendenti dai componenti logici (titolo, paragrafo, tabella, …)**

HTML: problema sugli stili

- **è diventato sempre più difficile creare siti Web dove il contenuto dei documenti HTML è chiaramente separato dalle regole per la loro formattazione**
- **i siti sono diventati insiemi molto complessi in cui si vuole mantenere un aspetto grafico omogeneo**
- **le regole di formattazione applicate in maniera “selvaggia” rendono difficile la manutenzione del sito**

La soluzione: CSS

- **Cascading Style Sheets**
- **introdotti dal World Wide Web Consortium (W3C) nella specifica HTML 4.0 ed ora usati anche per XHTML e XML**
- **i fogli di stile descrivono come visualizzare i componenti logici di un documento:**
  - il titolo di primo livello
  - lo sfondo della pagina
  - …
- **in questo modo si separano nettamente il contenuto ed il formato di presentazione**

CSS: definizione

- **CSS1 (dicembre 1996)**
  - http://www.w3.org/TR/REC-CSS1
  - 50 proprietà
- **CSS2 (maggio 1998) e CSS2.1 (luglio 2007)**
  - http://www.w3.org/TR/CSS2/ (vari errori)
  - http://www.w3.org/TR/CSS21/
  - aggiunge altre 70 proprietà a CSS1 (media-specific style-sheet – es. per stampanti e dispositivi audio – font scaricabili, tabelle, posizionamento di elementi)
- **CSS3 in corso di sviluppo**

CSS: risorse

- **CSS home page at W3C**
  - http://www.w3.org/Style/CSS/
- **tutorial ed altre risorse**
  - http://www.w3.org/Style/CSS/learning
- **servizio automatico di validazione**
  - http://jigsaw.w3.org/css-validator/

CSS

- **descrizione normalmente salvata in file di testo esterni al documento HTML**
  - file con estensione ".css"
- **file referenziato all'inizio del documento HTML**
- **lo stesso file CSS può essere incluso da molte pagine HTML**
  - è possibile cambiare l'aspetto di un intero sito modificando un solo file!
- **possibile – ma sconsigliato – definire lo stile in-line:**
  - per un singolo file
  - per singoli tag all'interno di uno specifico file

Sintassi CSS

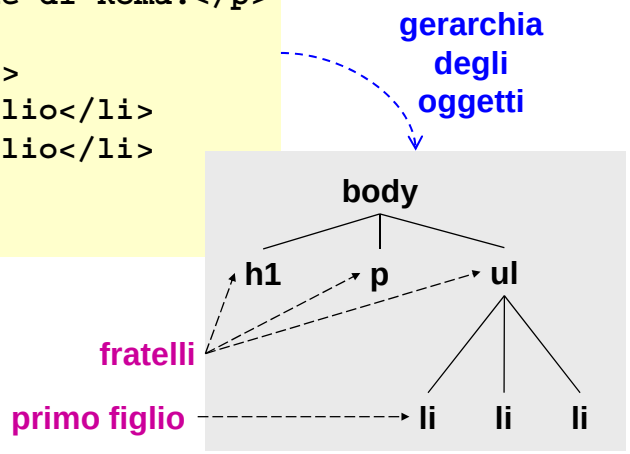
- **la sintassi CSS è costituita da 3 parti**
  - un selettore, una proprietà ed un valore
    - selector { property : value ; ... }
  - il selettore è il nome di un tag, di una classe o un identificatore unico
    - tag_name
    - [tag_name].class
    - [tag_name]:pseudo-class
    - #unique_id
  - il nome del tag può essere omesso (o si può usare *) per indicare qualunque tag per una classe

Sintassi CSS - esempi

```
/* tag */
body { background-color: gray }
p { font-family: "sans serif" }

/* tag.classe */
p.centrato { text-align: center; color: green }

/* classe */
.warning { color: red }

/* lista di tag */
h1, h2, h3 {color: green }

/* tag:pseudo-classe */
a:hover { background: aqua }

/* #id */
#footer { font-style: italic }
```

Visione gerarchica di HTML

```
...
<body>
  <h1>I Re di Roma</h1>
  <p>I primi tre Re di Roma:</p>
  <ul>
    <li>Romolo</li>
    <li>Numa Pompilio</li>
    <li>Tullo Ostilio</li>
  </ul>
</body>
```



Ereditarietà gerarchica

- le proprietà assegnate ad un elemento sono automaticamente ereditate da tutti i suoi discendenti
- possibile però fornire stile specifico per un discendente
- si applica sempre la regola più specifica
- esempio: per impostare il tipo di font di una pagina HTML è meglio specificare lo stile del BODY invece che di tutti gli elementi specifici
- eccezione: lo sfondo (background)

Selettori gerarchici

- **possibile specificare selettori in modo gerarchico per condizionare l'applicazione delle proprietà solo a casi specifici**
- **E1 E2**
  - elemento E2 quando discende da E1
- **E1 > E2**
  - elemento E2 quando è figlio di E1
- **E1 + E2**
  - elemento E2 quando immediatamente preceduto da un fratello E1
- **E:first-child**
  - elemento E quando è il primo figlio del suo genitore

Selettori condizionali

- **E [A]**
  - elemento E con un valore assegnato all'attributo A
- **E [A="Val"]**
  - elemento E con valore Val assegnato all'attributo A
- **E : lang(L)**
  - elemento E nella lingua L (es. "it", "de", "fr", "en-uk")
- **E : focus**
  - elemento E quando ha il "focus" (es. è attivo un campo per inserire testo)

Sintassi dei commenti in CSS

- stessa sintassi del linguaggio C, ossia /* … */

```
/* questo e' un commento */
p
{
text-align: center;
/* questo e' un altro commento */
color: red
}
```

Richiamare un foglio di stile (link)

- ciascuna pagina HTML che si basa su un certo file di stile deve includerne il nome nella sezione <head> mediante il tag <link rel="stylesheet" …>
- il browser leggerà le definizioni nel file (dopo averlo scaricato) e formatterà opportunamente la pagina da visualizzare
- esempio:

```
<head>
  <link rel="stylesheet"
    type="text/css"
    href="polito.css">
  <title> . . .
</head>
```

Foglio di stile interno

- si può definire uno stile direttamente nella sezione <head> mediante il tag <style>
- opzione sconsigliata perché si perdono i vantaggi della condivisione dello stile tra più pagine
- utile solo per fare prove rapide (migrare poi lo stile ad un CSS esterno)

```
<head>
  <style type="text/css">
    body { background-color : gray }
  </style>
  . . .
</head>
```

Per i vecchi browser ...

- i browser ignorano i tag sconosciuti ma visualizzano ciò che è contenuto tra i tag a meno che non sia commentato
- i nuovi browser ignoreranno i segni di commento (perché nei fogli di stile i commenti sono diversi)

```
<head>
<style type="text/css">
<!--
  body {background-color: gray }
-->
</style>
. . .
</head>
```

Stile per singoli tag (in-line style)

- **quando usare un CSS sarebbe troppo complesso si consiglia comunque di usare la sintassi del CSS tramite l'attributo STYLE nei singoli tag**
- **esempio:**

```
<!-- vecchio HTML -->

Prof. Antonio
<b>Lioy</b>
```

```
<!-- CSS-like -->

Prof. Antonio
<span style="font-weight:bold">Lioy</span>
```

Definizioni multiple

- **nel caso di definizione multipla dello stile di un elemento, il browser decide lo stile di tale elemento considerando "in cascata" tutte le definizioni con la seguente priorità:**
  - impostazioni sul tag (alta priorità)
  - foglio di stile interno
  - foglio di stile esterno (bassa priorità)

Definizioni multiple: esempio

- un foglio di stile esterno collegato alla pagina contiene la seguente definizione
  - `h1 {color: red; text-align: left}`
- mentre un foglio di stile interno alla pagina contiene:
  - `h1 {text-align: right}`
- la definizione risultante di `<h1>` per il browser sarà

```
color: red
text-align: right
```

Media-type

- possibile specificare che un foglio di stile si applica solo se la pagina è visualizzata su uno specifico supporto multimediale (o "media-type"):
  - parametro "media=…" del tag LINK
  - possibile elencare più media-type, separati da virgola
- esempio:

```
<head>
  <link rel="stylesheet" type="text/css"
    href="polito_stampa_A4.css" media="print">
    . . .
</head>
```

Media-type noti

- **definiti in CSS-2.1**
  - all (è il default)
  - braille (= dispositivo tattile Braille)
  - embossed (= stampante Braille a pagina)
  - handheld (= piccolo schermo, banda limitata)
  - print (stampante a pagina, pre-print view)
  - projection
  - screen
  - speech (in CSS2 era "aural")
  - tty (stampante a spaziatura fissa; vietato usare "px")
  - tv (=bassa risoluzione + audio)

Media-type e media-group

media-type	media group			
	paged, continuous	*visual, audio, speech, tactile*	*grid, bitmap*	*static, interactive*
braille	continuous	tactile	grid	both
embossed	paged	tactile	grid	static
handheld	both	visual, audio, speech	grid, bitmap	both
print	paged	visual	bitmap	static
projection	paged	visual	bitmap	interactive
screen	continuous	visual, audio	bitmap	both
speech	continuous	speech	N/A	both
tty	continuous	visual	grid	both
tv	both	visual, audio	bitmap	both

Includere un foglio di stile (@import)

- un CSS può importare le regole di altri CSS in modo assoluto o condizionato al media-type
- sintassi

  @import: url ("*URI_CSS*") [*media1* , *media2* , ...] ;
- da inserirsi prima di qualunque regola di formattazione
- esempio (all'interno di un file/sezione CSS):

```
@import url("poli_general.css");
@import url("poli_stampa.css") print;
body { color: blue }
. . .
```

CSS: definizione dell'attributo *class*

- è possibile definire stili differenti per uno stesso elemento di HTML usando l'attributo "class"
- esempi:
  - un paragrafo "normale" allineato a sinistra
  - un paragrafo "citazione" centrato e corsivo
  - una classe "centrato" da usarsi al posto di

```
p.normale { text-align: left }
p.citazione {
  text-align: center; font-style: italic }
/* sconsigliato: non e' una classe "logica" */
.centrato { text-align: center }
```

HTML: uso dell'attributo *class*

```
<p class="normale">
Il poeta ha scritto:
</p>
<p class="citazione">
Nel mezzo del cammin di nostra vita ...
</p>
```

Il poeta ha scritto:

Nel mezzo del cammin di nostra vita ...

CSS: unità di lunghezza

- **relative:**
  - "em" = altezza della lettera "M" nel font corrente
  - "ex" = altezza della lettera "x" nel font corrente
  - "px" = dimensione di un pixel
- **assolute:**
  - "in" = inch (25.4 mm)
  - "cm" o "mm"
  - "pt" = punto tipografico (1/72 di inch)
  - "pc" = pica (12 pt)
- **N.B. "em" ed "ex" sono riferite al font del genitore se usate per definire la dimensione di un font**

CSS: colori

- **alcuni colori predefiniti ed accessibili per nome:**

  Black, White, Gray, Silver,
  Yellow, Red, Purple, Fuchsia, Maroon,
  Green, Lime, Olive, Aqua, Teal, Blue, Navy

- **altri colori specificabili mediante il loro codice RGB esadecimale, decimale o percentuale**
- **esempi (equivalenti a "red"):**
  - rgb (255, 0, 0)
  - rgb (100%, 0%, 0%)
  - #ff0000
  - #f00 /* abbreviazione per due cifre hex uguali */
- **N.B. stessi nomi di HTML ma formato RGB diverso**

CSS: background (I)

- **background-color: *colore* | inherit | transparent**
  - "transparent" è il default
- **background-image: url("*image_URI*")**
  - URL dell'immagine
  - si suggerisce di specificare anche un colore da usarsi:
    - se non si riesce a caricare l'immagine
    - per contornare l'immagine
    - visibile nelle trasparenze dell'immagine

CSS: background (II)

- **background-repeat: repeat | repeat-x | repeat-y | no-repeat | inherit**
  - copie multiple dell'immagine per coprire tutto lo sfondo oppure solo in orizzontale o verticale
- **background-position: *posizione_h posizione_v***
  - ogni posizione può essere una keyword, una lunghezza (distanza da angolo in alto a sinistra) o una percentuale (della dimensione del contenitore)
  - keyword (orizzontale): left, center, right
  - keyword (verticale): top, center, bottom
  - se c'è una sola keyword la seconda si assume essere "center"

CSS: background (III)

- **background-attachment: fixed | scroll | inherit**
  - immagine in posizione fissa o segue l'eventuale scorrimento del contenuto
- **background: *color image_URI repeat attachment position* | inherit**
  - formato sintetico per tutte le varie specifiche
  - non importa l'ordine perché ogni elemento ha una sintassi diversa (possibile anche omettere elementi)
  - esempio (sfondo grigio, immagine centrata, segue il contenuto)

```
background: gray url("polito_logo.jpg")
  scroll center center no-repeat
```

CSS: text properties (I)

- **color: *colore***
- **text-align: left | right | center | justify**
- **text-transform: none | capitalize | uppercase | lowercase**
- **text-decoration: none | underline | overline | line-through | blink**
- **text-indent: *lunghezza* | *percentuale***
  - possibili valori negativi (testo sporge a sinistra)

CSS: text properties (II)

- **line-height: normal | *fattore* | *lunghezza* | *percentuale***
  - il valore del fattore moltiplicativo viene ereditato tale e quale mentre per la percentuale si eredita il valore numerico risultante
  - esempi:
    - { line-height: 1.2; font-size: 10pt } = i figli ereditano per l'altezza delle loro righe un fattore 1.2 da applicarsi alla dimensione del loro font
    - { line-height: 120%; font-size: 10pt } = i figli ereditano per l'altezza delle loro righe un valore fisso di 12pt

CSS: font properties (I)

- **font-style: normal | italic | oblique**
  - N.B. oblique sottilmente diverso da italic
- **font-weight: normal | bold | 1…1000**
  - il valore numerico indica la saturazione del nero
- **font-variant: normal | small-caps**
- **font-stretch: ultra-condensed | extra-condensed | condensed | semi-condensed | normal | semi-expanded | expanded | extra-expanded | ultra-expanded**
  - raramente disponibile su UA

CSS: font properties (II)

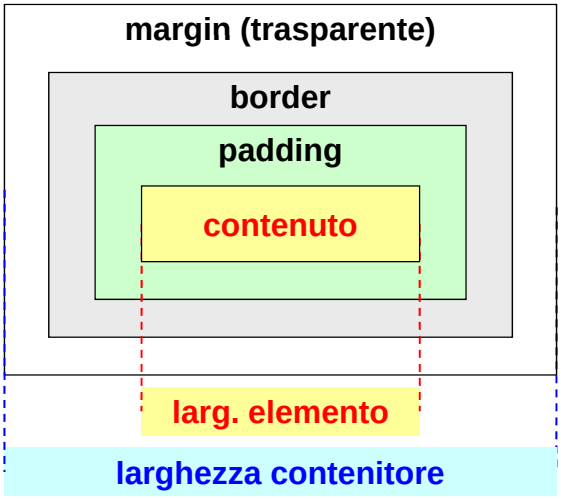
- **font-size: *dimensione_del_font***
  - valore assoluto (pt, mm, …)
  - valore relativo (em, ex, …)
  - percentuale
  - scala assoluta (1…7, normale=3) o relativa (+/– N)
  - xx-small x-small small medium large x-large xx-large (notare che manca “normal” e la scala non è definita)
- **note:**
  - i valori relativi e le percentuali sono riferite al font dell'elemento genitore (può essere il default dell'UA)
  - preferire valori relativi o percentuali, evitare valori assoluti (eccezione: media a dimensione fissa/nota)

CSS: font properties (III)

- **font-family: *lista_di_nomi_di_font***
  - lista ordinata (Courier, “Courier New”, …)
  - UA sceglie first match coi font disponibili localmente
  - utile specificare un font “generico” in ultima posizione
  - occhio alla compatibilità tra i font alternativi elencati
- **font generici (e compatibilità):**
  - serif (times, “times new roman”, palatino, ...)
  - sans-serif (arial, helvetica, verdana, ...)
  - monospace (courier, “courier new”, fixed, …)
  - cursive (“comic sans MS”, florence, ...)
  - fantasy (impact, oldtown, ...)

CSS: modello di formattazione

- **ogni blocco è visto come un contenitore (box)**



CSS: dimensioni dei contenitori

- **width: *lunghezza* | *percentuale* | auto | inherit**
- **min-width: *lunghezza* | *percentuale* | inherit**
- **max-width: *lunghezza* | *percentuale* | none | inherit**
  - larghezza (fissa, minima, massima) del contenuto
- **height: *lunghezza* | *percentuale* | auto | inherit**
- **min-height: *lunghezza* | *percentuale* | inherit**
- **max-height: *lunghezza* | *percentuale* | none | inherit**
  - altezza (fissa, minima, massima) del contenuto

- **i vari browser hanno default diversi per margin, border e padding – specificarli esplicitamente!**

CSS: dimensioni margini e padding

- **margin-left, margin-right, margin-top, margin-bottom: auto | *lunghezza* | *percentuale***
  - la percentuale è riferita alla larghezza del genitore
- **margin: *m_top_bottom_left_right***
  margin: *m_top_bottom m_left_right*
  margin: *m_top m_right_left m_bottom*
  margin: *m_top m_right m_bottom m_left*
  - formato sintetico (in quattro versioni) per le dimensioni dei margini
  - corrispondenza tra numero parametri e margini seguita anche per i bordi
- **padding-left, padding-right, padding-top, padding-bottom, padding: auto | *lunghezza* | *percentuale***

CSS: bordi dei contenitori

- **border-top-style, border-right-style, border-bottom-style, border-left-style, border-style: none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset**
  - stile di uno specifico bordo o forma sintetica
- **border-top-width, border-right-width, border-bottom-width, border-left-width, border-width: thin | medium | thick | *lunghezza***
  - dimensione di uno specifico bordo o forma sintetica
- **border-top-color, border-right-color, border-bottom-color, border-left-color, border-color: *colore* | transparent | inherit**
  - colore di uno specifico bordo o forma sintetica

CSS: formati sintetici per i bordi

- **border-top, border-right, border-bottom, border-left, border: [*width*] [*style*] [*color*]**
  - nella forma sintetica "border" si può specificare un unico valore uguale per tutti i bordi
- **esempio:**

```
border: medium ridge red
```

CSS: link properties

- **per definire lo stile dei link nei vari stati si usano pseudo-classi:**
  - :link (link non visitato)
  - :visited (link visitato)
  - :active (link su cui viene fatto click)
  - :hover (link puntato dal mouse)
- **oppure si possono associare classi / stili al tag A (indipendenti dallo stato)**
- **esempi:**

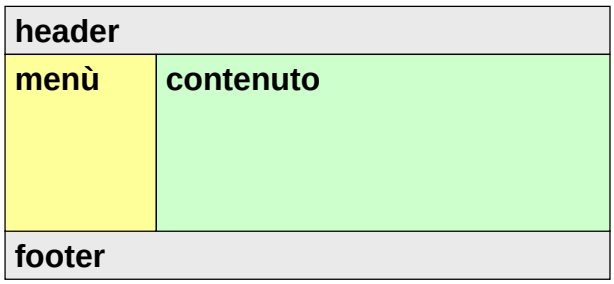
```
a:link { color: blue }
a:hover { background: aqua }
a.semplice { text-decoration: none }
```

CSS: layout grafico

- **posizionamento dei vari contenitori (box)**
- **float: left | right | none**
  - contenitore mobile
  - posizionamento orizzontale rispetto al padre
- **clear: left | right | both | none**
  - quali lati di un contenitore NON possono essere adiacenti a contenitori floating precedenti
- **N.B. molte altre istruzioni, modello molto complesso, leggere lo standard**

Un template standard

- **per pagine con header, footer, menù e contenuto**
- **moderno e standard:**
  - usa CSS per la formattazione
  - usa DIV per il layout
- **ispirato da http://friendlybit.com**



Template: parte HTML (I)

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"
  "http://www.w3.org/TR/html4/strict.dtd">
<html lang="it">
<head>
  <meta http-equiv="Content-Type"
    content="text/html; charset=ISO-8859-1">
  <meta name="Author" content="A.Lioy">
  <title>Titolo della pagina</title>
  <meta name="Keywords" content="web tutorial">
  <link rel="stylesheet"
    href="polito.css" type="text/css">
</head>
<body>
  <div id="header">
    <h1>Titolo della pagina</h1>
  </div>
```

Template: parte HTML (II)

```
  <div id="navigation"> <h2>Men&ugrave;</h2>
    <ul>
      <li><a href="p1.html">First</a></li>
      <li><a href="p2.html">Second</a></li>
    </ul>
  </div>
  <div id="content"> <h2>Contenuto</h2>
    <p>Contenuto della pagina.</p>
  </div>
  <div id="footer"> <h2>Autore</h2>
    <p>Nome dell'autore e info per contattarlo.</p>
  </div>
</body>
</html>
```

Template: CSS

```
body { ... }

div { border: 1px solid Black; padding: 1ex; }

h1, h2, h3, h4, h5, h6 { margin: 0; }

#navigation { float: left; width: 10ex; }
#navigation ul { list-style: none;
  margin: 0 0 0 0; padding: 0 0 0 0; }

#content { margin-left: 12ex; }

#footer { clear: both; }
```