

Sapienza Università di Roma  
corso di laurea in Ingegneria informatica e automatica

# **Linguaggi e tecnologie per il Web**

a.a. 2023/2024

## **Parte 2 CSS**

Lorenzo Marconi

# Fogli di stile

- In HTML non c'è separazione tra **forma** e **contenuto** del documento
- Fogli di stile o Cascading Style Sheets (CSS) = estensione di HTML introdotta da Internet Explorer 3
- Esempio: tag `<font>`
- I CSS si occupano di gestire la formattazione del documento esternamente al documento stesso
- Standard W3C (CSS1 e CSS2)

# Tipi di fogli di stile

- HTML permette di utilizzare diversi linguaggi per fogli di stile
- Linguaggio standard W3C per i fogli di stile: CSS (Cascading Style Sheets)
- `text/css`

# Tipi di CSS

- CSS in linea
  - agiscono su singole istanze di testo all'interno della pagina
- CSS incorporati
  - agiscono in modo globale su un singolo documento
- CSS esterni
  - agiscono in modo globale su insiemi di documenti

# CSS in linea

Agiscono su singole istanze di testo all'interno della pagina

Esempio:

```
<DIV STYLE="font-size:18px;  
    font-family:arial; color:red">  
    Questa parte di testo ha colore rosso  
</DIV>
```

Marcature più usate:

- senza “semantica” (<DIV> o <SPAN>)
- ricorda: `style` è un attributo globale!

# CSS in linea: pro e contro

- **Semplici** da usare (si possono considerare una “estensione” di HTML)
- **Difficili** da mantenere
- Modificano il contenuto del testo
- Non rispondono all’esigenza di separazione tra forma e contenuto
- È opportuno usarli in modo molto circoscritto

# CSS incorporati

- Agiscono in modo globale su un singolo documento HTML
- Vanno scritti nella parte intestazione del documento (tra `<HEAD>` e `</HEAD>`)
- Permettono di dare opzioni di formattazione **globale** ai tag (e quindi al documento)

# CSS incorporati: esempio

```
<HTML>
<HEAD>
<style type="text/css">
  H1 { font-size:17px; color:green; }
  H2 { font-family:arial; color:red; }
</style>
</HEAD>
<BODY>
  <H1>Tutti gli H1 sono di colore verde</H1>
  <H2>Tutti gli H2 sono di colore rosso</H2>
</BODY>
</HTML>
```



# CSS incorporati: sintassi

- Ogni dichiarazione CSS è una coppia **selettore – assegnazione di proprietà**
- Ad esempio:

```
H1 {font-size:17px; color:green;}
```

– H1 è il selettore

– {font-size:17px; color:green;} è  
l'assegnazione delle proprietà

# CSS incorporati: sintassi

```
H1 {font-size:17px; color:green }
```

```
H2 {font-family:arial; color:red }
```

- gli attributi sono inseriti tra parentesi graffe
- al posto del segno = vengono usati i due punti
- gli attributi composti da più termini sono separati da un trattino (kebab-case)
- quando un attributo è considerato proprietà di un oggetto i trattini si eliminano e le iniziali dei termini diventano maiuscole (camelCase)
- **esempio:** `font-style` diventa `fontStyle`

# Il tag <style>

```
<style type="text/css" media="...">
```

- L'attributo `TYPE` del tag `<STYLE>` definisce il **linguaggio** in formato MIME del foglio di stile
- `MEDIA` permette di specificare una **media query**
- Se l'attributo `TYPE` viene omissso, il browser lo identifica di default con `text/css`

**Attenzione:** non confondere il *tag* `<style>` con l'*attributo* `style` usato nei CSS in linea

# CSS incorporati: pro e contro

- I CSS incorporati permettono di configurare uniformemente la formattazione del testo (cioè l'aspetto dei tag) in un documento
- I CSS incorporati sono definiti al di fuori del corpo del documento, e permettono così la **separazione** tra contenuto e forma
- Tuttavia, i CSS incorporati sono inadatti a trattare in modo uniforme **insiemi di documenti** (ad esempio un sito web)

# CSS esterni

agiscono in modo **globale** su insiemi di documenti:

- gli stili dei singoli marcatori vengono raggruppati in un documento separato (file distinto dai documenti)
- ogni documento “richiama” gli stili (con un opportuno comando)
- una modifica sul file di stili genera automaticamente la stessa modifica su tutti i documenti che lo richiamano

# CSS esterni

Esempio: file `stile.css`:

```
H1 {font-size:17px; color:green}  
H2 {font-family:arial; color:red}
```

contiene gli stili che si vogliono definire

# Collegamento ad un CSS esterno

Collegamento ad un CSS esterno in un documento:

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

- Il tag `<link>` identifica un file esterno al documento HTML
- L'attributo `rel` indica il tipo di file collegato (stylesheet)
- L'attributo `href` richiama il percorso e il nome del file esterno

# Vantaggi dei CSS esterni

- Permettono la separazione tra contenuto e forma
- Permettono di gestire insiemi di documenti
- Massima flessibilità di utilizzo
- Riutilizzabilità degli stili
- Possibilità di **fondere** insieme più stili (*cascading*)



# Attributi di stile per il testo

- font-family (serif / sans serif / cursive ....)
- font-size
  - medium / xx-small / x-small / small / large / x-large / xx-large / smaller / larger / [valore personalizzato]
  - punti (pt) / pixel (px) / centimetri (cm) / pollici (in) / percentuale (%), ...
- font-style (normal / italic / oblique)
- font-variant (normal / small-caps)
- font-weight
  - normal / bold / bolder / lighter / [valore personalizzato]
- text-decoration
  - none / overline / underline / line-through

# Esempio

Per i CSS in linea:

```
<A style="text-decoration:none"  
  href="#a1">ciao </a>
```

Per i CSS incorporati:

```
<style>  
  A {text-decoration: none}  
</style>
```

# Versioni di CSS

- Il W3C ha rilasciato varie versioni di CSS:
  - CSS 1 (1996)
  - CSS 2 (1998)
  - CSS 2.1 (2011)
  - CSS 3 (2011-2014)
- Tra le differenze principali delle varie versioni, si segnala il numero di proprietà definite e quindi assegnabili:
  - CSS1 = 53 proprietà
  - CSS2 = 122 proprietà
  - CSS3 = 339 proprietà

# Selettori CSS

- Fino ad ora abbiamo visto selettori CSS corrispondenti a nomi di elementi HTML (che selezionano tutti gli elementi del documento HTML con quel nome)
- In realtà si possono costruire selettori molto più complessi

# Selettori CSS

- Selettore universale: \*
- Selettore di nome: *nome-elemento* (es. H1)
- Selettore di classe: *.nome-classe*
- Selettore di classi multiple:  
*.nome-classe1.nome-classe2...*
- Selettore di id: *#nome-id*
- Selettore di discendenti (sequenza)
- Unione di selettori (,)
- Selettore di figli (>)
- Selettori di fratelli (+), (~)

# Selettori CSS

- Selettori di pseudo-classi:
  - E:link
  - E:visited
  - E:hover
  - E:focus
  - E:first-child
  - E:root
  - E:nth-child()

# Selettori CSS

- Selettori di pseudo-classi:
  - E:nth-last-child()
  - E:last-child
  - E:only-child
  - E:nth-of-type()
  - E:nth-last-of-type()
  - E:first-of-type
  - E:last-of-type
  - E:only-of-type

# Selettori CSS

- Selettori di pseudo-classi:
  - E:not
  - E:empty
  - E:target
  - E:enabled
  - E:disabled
  - E:checked



# Selettori CSS

- Selettori di pseudo-classi:
  - E:default
  - E:valid
  - E:invalid
  - E:in-range
  - E:out-of-range
  - E:required
  - E:optional
  - E:read-only
  - E:read-write

# Selettori CSS

- Selettori di pseudo-elementi:
  - ::first-letter
  - ::first-line
  - ::before
  - ::after

# Selettori CSS

- Selettori di attributi:
  - E[attribute]
  - E[attribute=value]
  - E[attribute~=value]
  - E[attribute|=value]
  - E[attribute^=value]
  - E[attribute\$=value]
  - E[attribute\*=value]

# Classi

- A tutti gli elementi è possibile assegnare l'attributo `CLASS`
- Questo attributo permette ai fogli di stile di trattare singole occorrenze o sottoinsiemi di occorrenze dello stesso elemento o di elementi diversi

# Classi: esempio

```
<STYLE>
  H2.top { font-family:verdana; font-size:15px;
    font-weight:bold; font-style:normal; }
  H2.bottom { font-family:arial; font-size:10px;
    font-weight:bold; font-style:italic; }
</STYLE>

<H2 class="top">Testo della pagina</H2>
<H2 class="top">Altro testo della pagina</H2>
...
<H2 class="bottom">Testo della pagina</H2>
<H2 class="bottom">Altro testo della pagina</H2>
```

# Pseudoclassi

- esempio: elementi anchor:
  - quando visitati costituiscono una pseudoclasse `visited`, quando attivi `active` e quando non visitati `link`
- viene specificata all'interno dello stile seguita dai due punti
- esempio:

```
<STYLE>  
  A:link { text-decoration: none; }  
  A:visited { text-decoration: none; }  
</STYLE>
```

# Identificatori

- A tutti gli elementi è possibile assegnare l'attributo ID: il suo valore identifica univocamente quell'elemento all'interno del documento
- Questo attributo permette ai fogli di stile di trattare singole occorrenze di elementi

# Identificatori

## Esempio:

...

```
<STYLE>
```

```
  #mioelem {font-weight:bold; }
```

```
</STYLE>
```

...

```
<p id="mioelem">testo 1</p>
```

```
<p>testo 2</p>
```

...

**Il paragrafo "testo 1" verrà visualizzato in neretto**



# Proprietà CSS nel DOM

- Nel DOM ogni elemento HTML ha la proprietà `style`, che contiene tutte le proprietà CSS dichiarate per quell'elemento
- Il valore di tale proprietà `style` è un oggetto `CSSStyleDeclaration`

# Proprietà CSS nel DOM

- Proprietà dell'oggetto `CSSStyleDeclaration`:
  - `length`
  - `cssText`
  - `parentRule`
- Metodi dell'oggetto `CSSStyleDeclaration`:
  - `item(n)`
    - restituisce il nome della proprietà CSS di indice `n` dell'elemento
  - `getPropertyValue(nomeProprietàCSS)`
  - `setProperty(nomeProprietàCSS)`
  - `removeProperty(nomeProprietàCSS)`
  - `getPropertyPriority(nomeProprietàCSS)`

# Proprietà CSS nel DOM

- Nel DOM le proprietà CSS possono essere accedute come (sotto)proprietà della proprietà style dell'elemento:
- Esempio:  

```
document.getElementById("abc").style.color="white";
```

assegna il colore bianco all'elemento con id uguale ad "abc"
- Altro esempio:  

```
document.getElementsByTagName("p")[0].style.text-align="right";
```

assegna allineamento destro al primo elemento paragraph (p) del documento

# Proprietà CSS nel DOM

- Se non si conoscono a priori le proprietà CSS definite per un elemento HTML, si possono estrarre a runtime
- Esempio:

```
var s=new Array();  
var x=document.getElementById("abc");  
for (var i=0;i<x.style.length;i++) s[i]=x.style.item(i);
```

memorizza nell'array s l'elenco delle proprietà CSS definite per l'elemento con id uguale ad "abc"

# Riferimenti

- Raccomandazioni ufficiali W3C:

[www.w3.org/Style/CSS](http://www.w3.org/Style/CSS)

- Tabella riassuntiva delle proprietà CSS rispetto alle varie versioni dello standard:

[meiert.com/en/indices/css-properties](http://meiert.com/en/indices/css-properties)

- Tutorial CSS sul sito W3Schools:

[www.w3schools.com/css](http://www.w3schools.com/css)