

INFORMATICA PER LA COMUNICAZIONE



CSS

CSS

Fogli di stile



Prima dei CSS

- Fino all'introduzione dei CSS:
 - ▣ Codice HTML per definire struttura della pagina e grafica
 - ▣ Codice HTML non accessibile per:
 - Ai motori di ricerca
 - Ai browser testuali



CSS

CSS (*Cascading Style Sheets*): standard per la definizione della **formattazione**

- Evitare utilizzo degli attributi di formattazione HTML
- Separare definizione di
 - ▣ **Struttura** → codice HTML
 - ▣ **Formattazione** → fogli di stile CSS



CSS: formattazione

- CSS definiscono la formattazione della pagina
 - ▣ **Carattere:** font, colore, dimensione, ...
 - ▣ **Blocchi:** dimensioni, sfondi, bordi

Esempio CSS:

```
p { font-family: Arial; color:  
blue; font-size: 16px;}
```



CSS: modalità di definizione

Fogli di stile possono essere:

1. **Esterni:** in file esterni (file .css)
2. ***Embedded:*** codice interno alla pagina, contenuto nell'intestazione
3. ***Inline:*** codice in linea col testo al livello di un singolo elemento



CSS: file esterno

- I file CSS sono in genere esterni e sono referenziati all'interno dell'head

```
<head>
<title> Esempio CSS </title>
<link rel="stylesheet"
      href="fogliostile.css">
</head>
```

Riferimento al file
CSS esterno

CSS per tag

Regole sintattiche

```
tag {proprietà1:valore1; proprietà2:valore2;...}
```

- **Proprietà:** specificano aspetti di formattazione da definire

```
h1, h2 {font-size: 14px;  
        font-family: arial;  
        color: rgb(255,255,255);  
        background: rgb(0,0,0);  
}
```

Codifica colori

- **Corpo del documento:**

- **Definizione colore:**

- **rgb(a,b,c)**

- **#RRGGBB**

Notazione per specificare la quantità di rosso, verde, blu

Es.

Grigio chiaro → *rgb(211,211,211)*

Nero → *rgb(0,0,0)*

Bianco → *rgb(255,255,255)*



Blocchi di testo

- Per delimitare un blocco
 - ▣ Applichiamo un bordo con l'attributo `style` (`border`)

```
<div style="border:1px solid black">  
<p>Div con un bordo nero</p>  
</div>
```



Font

- I caratteri utilizzabili si dividono in due categorie:
 - ▣ **Web safe:** caratteri più diffusi nei client
 - Arial, Helvetica, Verdana, Times New Roman, Georgia,...

```
font-family: Georgia, "Times New Roman", serif;
```

- ▣ **Esterni**



Font esterni

- I font possono essere incorporati dall'esterno:
servizi per incorporare font nei file CSS
 - ▣ **Semplificazione** nella progettazione CSS
 - ▣ Gestione licenze d'uso
 - ▣ **Google fonts:**

<https://fonts.google.com/>



Font esterni: comandi CSS

- Per incorporare un font di Google Fonts:

```
@import  
url(https://fonts.googleapis.com/css?family=Montserrat:400,400i,700);
```

font

corsivo

grassetto

- Per utilizzarlo:

```
font-family: 'Montserrat', sans-serif;
```

Font esterni: esempio

```
@import
url('https://fonts.googleapis.com/css?family=Montserrat:400,400i,700'
);

body {
font-family: 'Montserrat', sans-serif;
}
```

Laurea magistrale in Scienze della Comunicazione, Informazione, Editoria [Unibg](#)

:

- [Home](#)
- [Corso](#)

Presentazione del corso di laurea



Applicazione a cascata

- Nell'applicazione delle regole definite dai CSS vale il principio di **cascata**: risoluzione di regole conflittuali
 1. **Origine** della regola
 2. **Specificità** del selettore
 3. **Ordine** in cui compaiono nel foglio di stile



CSS: specificità

- Dato un certo elemento, viene applicato lo stile più **specifico**

CSS

```
div h1{  
color: blue;}  
  
p {  
color: red;}  
  
h1 {  
color: green;}
```

HTML

```
<div id="intestazione">  
<h1>  
Titolo dell'articolo  
</h1>  
</div>
```

Titolo dell'articolo



CSS con classe

- I tag possono essere associati a **classi**
- Attributo HTML *class*
 - ▣ Uno stesso tag, a seconda delle classe, può avere **caratteristiche differenti**
 - ▣ Usando `.class` posso applicare la stessa formattazione a più tag di una certa classe



CSS con classe - Esempi

CSS

```
div.avviso {  
color: rgb(0,0,255);  
background: rgb(255,0,0);  
}  
.notizia  
{  
color: rgb(0,255,0);  
background: rgb(255,255,0);  
}
```

HTML

```
<div class="avviso">  
Questo è un avviso  
</div>  
  
<div class="notizia">  
Questo è una notizia  
</div>
```

Questo è un avviso

Questa è una notizia



CSS con ID

- I tag possono essere associati a **id** così da identificare ogni singolo tag

CSS

```
.avviso{  
color: blue;  
background: red;}  
  
#intestazione {  
color: blue;  
font-size: 200%;}
```

HTML

```
<div id="intestazione">  
Esempio di utilizzo CCS  
</div>  
  
<p class="avviso">  
Questo è un avviso  
</p>
```

Esempio di utilizzo CCS

Questo è un avviso



CSS: Tipografia

Testo e pagina



Testo

- Inizialmente le proprietà del carattere erano controllate dal **browser**
- In `HTML2` il tag `font` permise il controllo delle proprietà del carattere
- Nelle versioni **CSS2** e **CSS3**: possibilità di impostare le proprietà del carattere dal **foglio di stile**



Testo

- Principali impostazioni formattazione del testo con CSS
 - ▣ `font-family`
 - ▣ `font-size`
 - ▣ `line-height`
 - ▣ `color`
 - ▣ `font-style`
 - ▣ `text-align`
 - ▣ ...



Dimensione del carattere

- Con la proprietà `font-size` possiamo specificare la **dimensione** del carattere:
 - ▣ Valore con unità di misura
 - Relative: `%`, `em`, `px`,...
 - Absolute: `in`, `cm`, `pt`,...

```
#intestazione {  
color: blue;  
font-size: 12px;}
```

```
body {  
color: blue;  
font-size: 1em;}
```



CSS: impaginazione

Posizionamento degli elementi



Posizionamento degli elementi

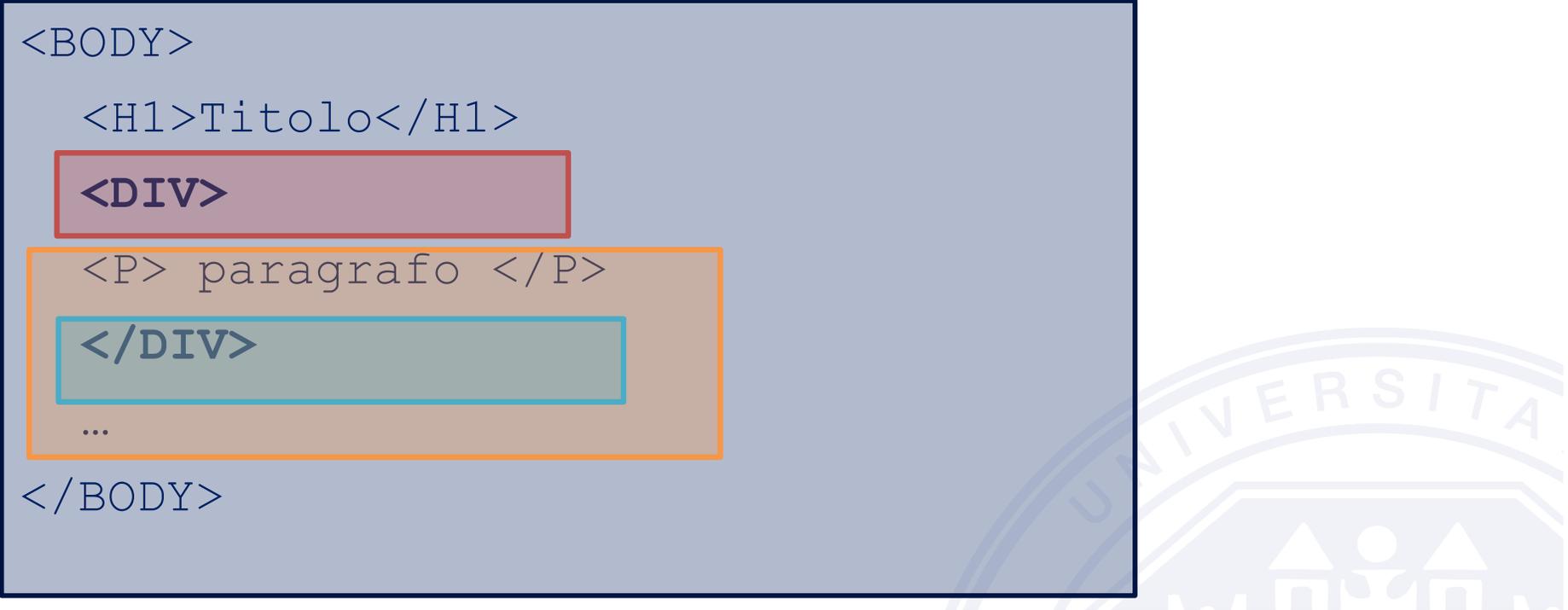
- CSS permettono di posizionare un elemento rispetto al contenitore:
 - ▣ Elementi **blocco** (es. `p` e `div`) definiscono un blocco, delimitano uno spazio
 - ▣ Elementi **inline** (es. `img` e `a`) posti in linea con altri elementi



CSS: elementi blocco

Il documento viene visto come una serie di blocchi

```
<BODY>
  <H1>Titolo</H1>
  <DIV>
    <P> paragrafo </P>
  </DIV>
  ...
</BODY>
```



Posizionare gli elementi blocco

- Un elemento **blocco** può avere un posizionamento
 - ▣ **Relativo** rispetto all'elemento che lo contiene
 - ▣ **Assoluto** rispetto alla pagina
- Inoltre un elemento può essere **fluttuante**
- Il **comportamento predefinito** prevede che i blocchi siano disposti uno dopo l'altro



Posizionamento relativo

- Gli elementi sono posti in successione **in verticale** (a meno che non siano fatti fluttuare)
- Per specificare questa caratteristica nel file CSS
position: relative



Posizionamento assoluto

Posizionamento assoluto: posizionamento rispetto al punto di riferimento della pagina

```
.modifica {  
position: absolute;  
top: 1px;  
left: 1px;  
z-index: 99;  
}
```



Posizionamento `float`

- Elementi **blocco**

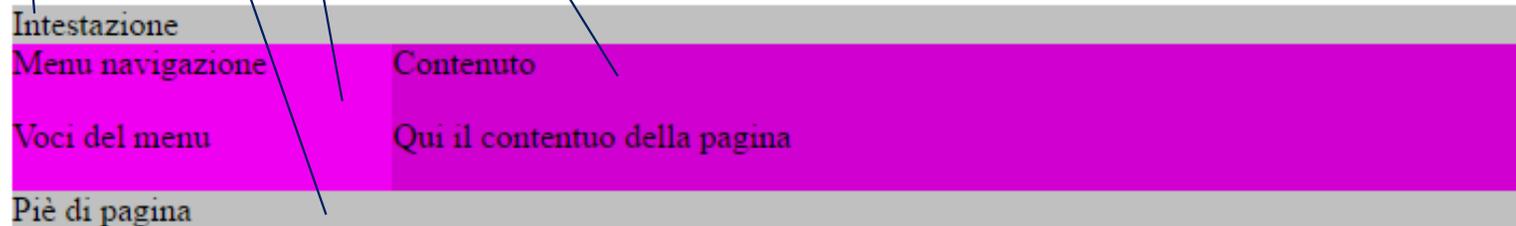
- ▣ Quando il comportamento non è modificato: elementi posti in successione **in verticale**

- ▣ Usando la proprietà **`float`** è possibile far *fluttuare* gli elementi nello spazio



Posizionamento - Esempio

- float elemento collocato con altri
- clear elemento occupa tutto lo spazio



Menu di navigazione

Costruiamo un menu di navigazione **orizzontale**



A horizontal navigation menu with a black background and white text. The menu items are 'Home', 'Didattica', and 'Organizzazione', separated by small gaps. A thin blue underline is visible above the 'Home' text.

Home Didattica Organizzazione



Menu di navigazione

```
nav{
  position:relative;
  color: rgb(255,255,255);
  background: rgb(0,0,0);
  height:60px;
}

ul {
  list-style-type:
  none;
  margin: 0;
  padding: 0;
}

li {
  float: left;
}

li a {
  color: rgb(255,255,255);
  display: block;
  text-align: left;
  padding: 14px 16px;
  text-decoration: none;
}
```



Posizionamento

- Parametri che influenzano il posizionamento di un elemento (div, span, etc...)
 - ▣ `border`: bordo di un elemento
 - ▣ `padding`: spazio tra bordo dell'elemento e contenuto
 - ▣ `margin`: spazio tra bordo dell'elemento ed elementi vicini

