

(X) HTML + CSS

Elementi Blocco vs elementi inline

Titolo 1

Primo paragrafo Lorem ipsum dolor sit amet, **consectetur** adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim **veniam**, quis nostrud exercitation ullam corper suscipit lobortis nisl ut aliquip ex ea commodo consequat.

Titolo 2

Secondo paragrafo Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut



laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad **minim** veniam, quis nostrud exercitation ullam corper suscipit lobortis nisl ut aliquip ex ea commodo consequat.

Cella 1	Cella 2
Cella 3	Cella 4

HTML vs XHTML

- Prologo distinto
- Tag denotati sempre in minuscolo
- Tutti i tag aperti devono essere chiusi
- I valori degli attributi vanno sempre inclusi fra coppie di virgolette. **Es.:** `class="pippo"`
- I tag di apertura privi di tag di chiusura si denotano nel seguente modo: `<tag />`

Es.: `
`

Il tag SPAN

- Utilizzato molto frequentemente in XHTML per applicare proprietà a porzioni di testo.

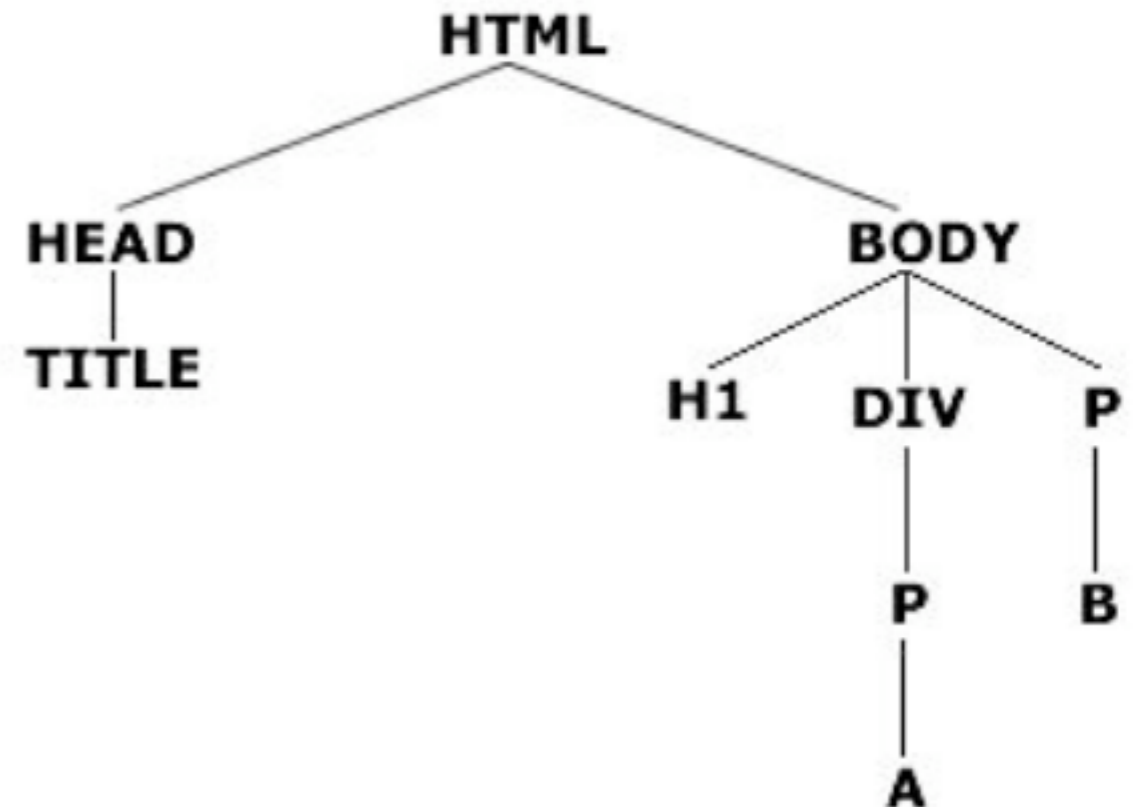
```
<p> questo è un  
    <span> blocco di testo </span>  
    incluso in un paragrafo  
</p>
```

Struttura di un documento (X)HTML

Codice (X)HTML

```
<html>
  <head>
    <title>
      Struttura del documento
    </title>
  </head>
  <body>
    <h1>Titolo</h1>
    <div> <p>Primo
      <a href="pagina.html">
        paragrafo
      </a>
    </p>
  </div>
  <p>Secondo
    <b>paragrafo</b>
  </p>
</body>
</html>
```

Albero



Relazioni:

- Genitore-Figlio
- Antenato-Discendente

CSS + (X)HTML

Codice (X)HTML

definisce la struttura del documento

Codice CSS

definisce la presentazione del documento (**foglio di stile**)

Includere un foglio di stile in un documento (X)HTML (I)

Modo I: collegare un foglio di stile esterno a un documento XHTML

Nell'intestazione (<head>... </head>) del documento (X)HTML specificare

```
<link rel="stylesheet" type="text/css"
      href="nome-stile-da-includere.css" />
```

Includere un foglio di stile in un documento (X)HTML (2)

Modo II: includere un foglio di stile in un documento XHTML

Nell'intestazione (<head>... </head>) del documento (X)HTML specificare

```
<style type="text/css">  
    ... regole di stile ...  
</style>
```


Includere un foglio di stile in un documento (X)HTML (3)

Modo III: fogli in linea ovvero modifiche puntuali delle proprietà dei tag di un documento XHTML

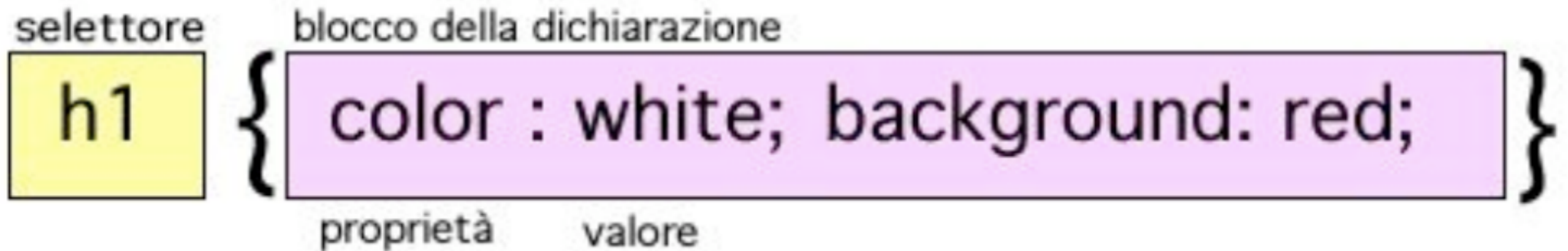
La dichiarazione avviene a livello dei singoli tag. La sintassi generica è la seguente:

```
<tag style="regole_di_stile"> ... </tag>
```

oppure

```
<tag style="regole_di_stile" />
```

Sintassi regole CSS



- Il **selettore** serve a definire la parte del documento cui verrà applicata la regola
- La **proprietà** definisce un aspetto dell'elemento da modificare attraverso un **valore** dato.
- **Ereditarietà** delle proprietà

Selettori: selettore di elementi

- È costituito da un singolo tag (X)HTML

```
h1 {color: #000000;}
```

```
p {background: white;  
  font: 12px Verdana, arial, sans-serif;}
```

```
table {width: 200px;}
```

Selettori: raggruppamento

- È costituito da una lista di elementi separati da virgole

```
h1, h2, h3 {color: #000000;}
```

equivale alle regole

```
h1 {color: #000000;}
```

```
h2 {color: #000000;}
```

```
h3 {color: #000000;}
```

Selettori: selettore universale *

- seleziona tutti gli elementi di un documento. È denotato con *

```
* {color: #000000;}
```

Selettori: selettore del discendente

- seleziona tutti gli elementi discendenti di un altro elemento nell'albero (X)HTML

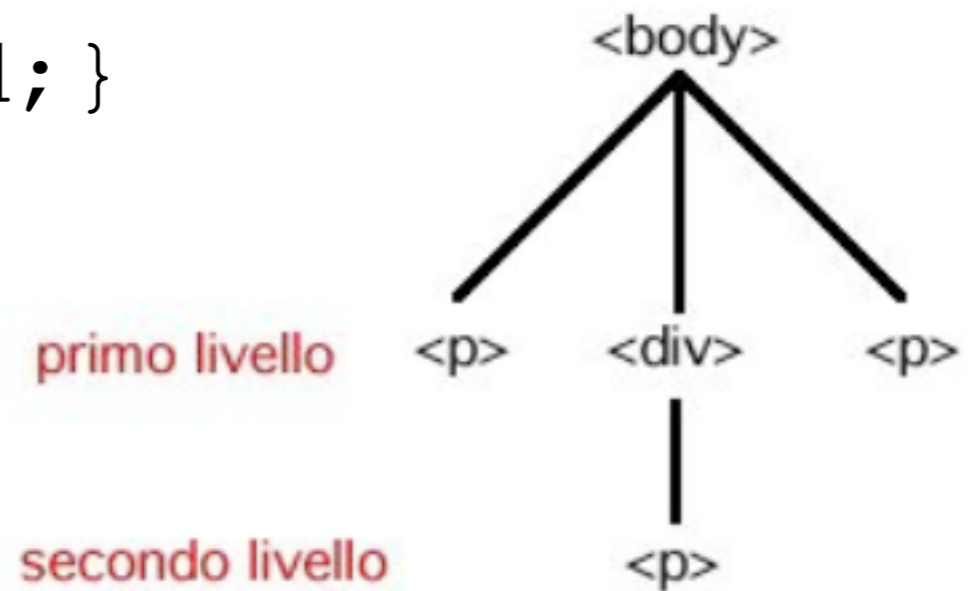
```
p strong {color: red;}
```

```
div p emph {color: green;}
```

Selettori: selettore del figlio

- seleziona tutti gli elementi figli di un altro elemento nell'albero (X)HTML
(discendenza diretta o di I livello)

```
body > p {color: red;}
```



Selettori: selettore dell'attributo

- seleziona gli elementi in base ai loro attributi di un altro elemento nell'albero (non molto supportato dai browsers)

```
a [ href ] {background: red;}
```

contiene attributo **href**

```
a [ href = "http://www.google.it" ] {color: red;}
```

contiene attributo **href** con valore **http://www.google.it**

```
img [ alt t= "foto" ] {margin: 10px;}
```

il valore dell'attributo **alt** contiene **foto**

```
img [ alt |= "figura" ] {margin: 10px;}
```

il valore dell'attributo **alt** inizia con la parola **figura**

Selettori: le classi

- permettono di applicare un insieme di proprietà a tutti gli elementi (X)HTML che le riferenziano

In (X)HTML si deve riferenziare la classe mediante attributo **class**

```
<h1 class="testorosso"> titolo </h1>  
<p class="testorosso"> blablabla </p>
```

Nel foglio di stile si deve definire la classe

```
.testorosso {color: red;  
              font: 14pt arial,sans-serif; }
```

Alcune pseudoclassi

```
a:link {color:red;} /* link non visitati */  
a:visited {color:blue;} /* link visitati */  
a:hover {color:yellow;} /* link sorvolati */  
a:active {color:lime;} /* link attivi */
```

Disporre le definizioni in questo ordine, altrimenti le regole a cascata possono non fare attivare alcune delle pseudoclassi

Alcuni pseudoelementi

Lo pseudoelemento **:first-letter** si applica alla prima lettera del contenuto di un elemento (e.g., un paragrafo)

```
p:first-letter { font-size:34pt; color:red; }
```

Lo pseudoelemento **:first-line** si applica alla prima linea del contenuto di un elemento (e.g., un paragrafo)

```
p:first-line { font-size:34pt; color:red; }
```

Selettori: identificatori

- un'identificatore permette di applicare un insieme di proprietà a **un singolo** elemento (X)HTML che lo riferenzia.

In (X)HTML si deve riferenziare l'identificatore mediante attributo **id**

```
<h1 id="sfondoverde"> titolo </h1>
```

Nel foglio di stile si deve definire la classe

```
#sfondoverde{background-color: green; }
```

Errore se aggiungo una linea di codice XHTML del tipo

```
<p id="sfondoverde"> blablabla </h1>
```

Gestione colore

- la proprietà **color** permette di selezionare il colore di primo piano

Valori possibili per **color**

black | navy | blue | maroon | purple | green | red |
teal | fuchsia | olive | gray | lime | aqua | silver |
yellow | white

oppure

Notazione rgb esecimale #RRGGBB

oppure

Notazione rgb rgb (RRR%, GGG%, BBB%)

Gestione sfondo

- la proprietà **background-color** definisce un'immagine da usare come sfondo

```
h1 {background-color:rgb(100%,0%,0%);}
```

- la proprietà **background-image** permette di selezionare il colore di sfondo

```
p {background-image:url(sfondo.gif);}
```

- la proprietà **background-repeat** permette di ripetere l'immagine di sfondo.
Valori: repeat, repeat-x, repeat-y, no-repeat.

```
p {background-repeat:repeat-y;}
```

Gestione sfondo

- la proprietà **background-position** permette di decidere la posizione dell'immagine di sfondo. **Valori:** percentuali, unità di misura, top, left, bottom, right.

```
body { background-image: url(back_400.gif);  
        background-repeat: no-repeat;  
        background-position: center center; }
```

Gestione sfondo

- la proprietà **background** riassume le proprietà precedenti

```
selettore {background: background-color  
background-image  
background-repeat  
background-attachment  
background-position;}
```

```
body { background: url(sfondo.gif)  
repeat-x  
fixed; }
```


Gestione testo

- la proprietà **font-family** definisce una lista di font utilizzabili

```
h1 {font-family: arial, Verdana, sans-serif;}
```

- la proprietà **font-size** definisce la dimensione del font

```
h1 {font-family: arial, Verdana, sans-serif;  
    font-size: 14mm; }
```

- la proprietà **font-weight** definisce il peso del grassetto. **Valori:** 100,200,300...,900

```
h1 {font-family: arial, Verdana, sans-serif;  
    font-weight: 900; }
```

Gestione testo

- la proprietà **font-style** lo stile del font.
Valori: normal, italic, oblique.

```
h1 {font-family: arial, Verdana, sans-serif;  
    font-style:italic; }
```

- la proprietà **line-height** definisce
l'interlinea. Valori assoluti o con unità di
misura.

```
p {font-family: arial, Verdana, sans-serif;  
    line-height: 1.5; }
```

- la proprietà **font** riassume le proprietà
precedenti

```
p {font: 700 12px/1.5 "Times New Roman", serif; }
```

Gestione testo

- la proprietà **text-indent** permette di indentare la prima riga di un testo. Valori: unità di misura, percentuale.

```
p {text-indent: 10mm; }
```

- la proprietà **text-align** permette allineare del testo. Valori: **left, right, center, justify**

```
p {text-align: right; }
```

- la proprietà **text-decoration** descrive il tipo di decorazione da aggiungere al testo. Valori: **underline, overline, line-through, none**

Gestione testo

- la proprietà **letter-spacing** permette di definire lo spazio fra i singoli caratteri.

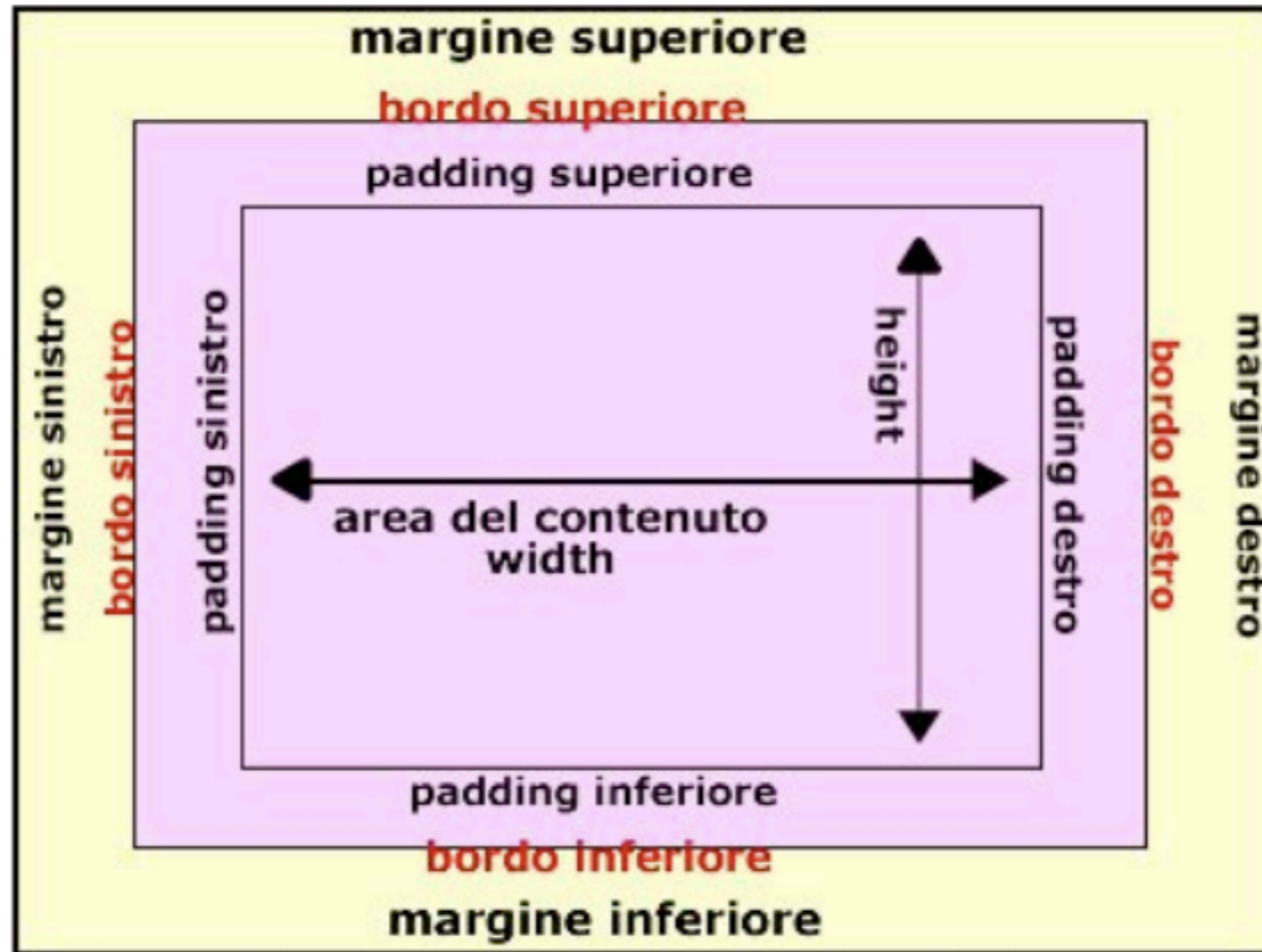
```
p {letter-spacing: 10mm;}
```

- la proprietà **word-spacing** permette di definire lo spazio fra le parole di un testo.

```
p {word-spacing: 3cm;}
```

- la proprietà **text-decoration** descrive il tipo di decorazione da aggiungere al testo. Valori: **underline, overline, line-through, none**

Gestione elementi blocco: box model



Gestione elementi blocco: box model

- Proprietà **margin-left, margin-right, margin-top, margin-bottom**
- Proprietà **margin**

Valori possibili (anche negativi) **auto, valore percentuale, valore con unità di misura**

```
div {margin-left: 50px;}
```

```
p {margin: 10px 5%;}
```

```
img {margin-left: auto; /* centra un'immagine */  
margin-right: auto;} /* rispetto al contenitore*/
```

Gestione elementi blocco: padding

- Proprietà **padding-left, padding-right, padding-top, padding-bottom**
- Proprietà **padding**

Valori possibili (solo maggiori o uguali a 0) **auto, valore percentuale, valore con unità di misura**

```
div {padding-left: 1cm;}
```

```
p {padding: 10mm 4px 3mm 6%;}
```

Gestione elementi blocco: i bordi

- Controllo del colore, spessore, stile dei singoli bordi via

```
border-top-color, border-top-style,  
border-top-width, border-bottom-color,  
border-bottom-style, border-bottom-width,  
border-right-color, border-right-style,  
border-right-width, border-left-color,  
border-left-style, border-left-width
```

Valori per lo stile del bordo:

```
none, hidden, dotted, dashed, solid, double, groove,  
ridge, inset, outset
```


Gestione elementi blocco: i bordi

- Controllo del colore, spessore, stile dei 4 bordi via

```
selettore { border-width: <valori>;  
            border-style: <valori>;  
            border-color: <valori>; }
```

```
div { border-width: 1px 2px 1px 2px;  
      border-style: solid;  
      border-color: black red black red; }
```

Gestione elementi blocco: i bordi

- Controllo del stesso colore, spessore, stile dei 4 bordi via

```
selettore {border: <spessore> <stile> <colore>;}
```

```
div { border: 10mm dotted blue;}
```

Gestione liste

- la proprietà **list-style-image** definisce l'immagine da utilizzare come marcatore delle voci della lista

```
ul {list-style-image: url (immagine1.gif);}
```

```
ol {list-style-image: url (immagine2.gif);}
```

```
li {list-style-image: url (immagine3.gif);}
```

Gestione liste

- la proprietà **list-style-position** definisce la posizione del marcatore di voce di lista rispetto al testo delle voci di lista. Valori: **outside, inside**
- la proprietà **list-style-type** definisce il tipo di marcatore delle voci di lista.

Valori: `disc, circle, square, decimal, decimal-leading-zero, lower-roman, upper-roman, lower-alpha, upper-alpha, lower-greek`

Gestione liste

- la proprietà **list-style** riassume le proprietà precedenti.

```
<selettore> {list-style: <valore tipo>  
                <valore posizione>  
                <valore immagine>; }
```

```
ul li {list-style: outside url(imamgine.gif); }
```

Gestione elementi

- la proprietà **display** cambia la natura di un elemento. Valori comunemente utilizzati: **block, inline**

```
img {display: block;}
```

Gestione elementi flottanti

- la proprietà **float** sposta un elemento sul lato sinistro/destro del suo blocco contenitore. Valori: **left, right, none**

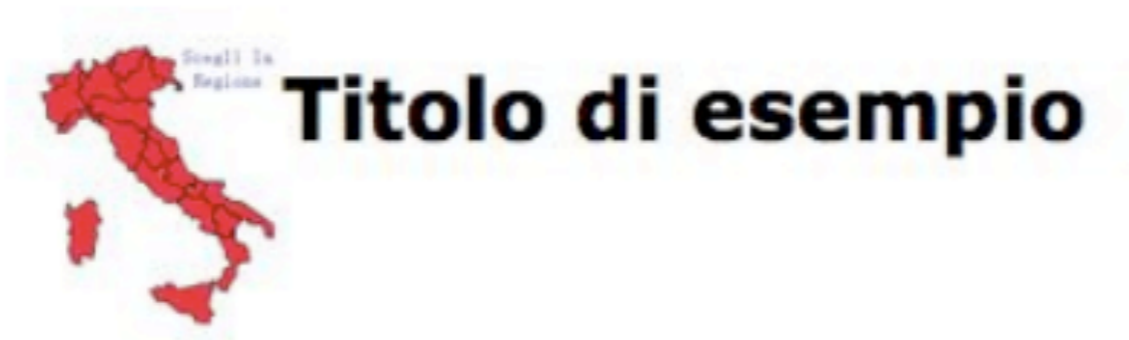
```
img {float: left;}
```

- la proprietà **clear** impedisce che a fianco di un elemento ci siano altri elementi con proprietà **float**. Valori: **left, right, both, none**

```
h1 {clear: left;}
```

Gestione elementi flottanti

```
img { float:left; }
```



Sotto invece abbiamo applicato al titolo questa classe:

```
h1.clear {  
clear : left;  
}
```

Come conseguenza impediamo all'immagine di stare al suo fianco sinistro.

