

## Indice

|                                                                |          |
|----------------------------------------------------------------|----------|
| <b>6. HTML (CONTINUA)</b> .....                                | <b>2</b> |
| 6.1 FOGLI DI STILE (CSS- CASCADING STYLE SHEET) .....          | 2        |
| 6.1.1 Regole dello stile.....                                  | 6        |
| 6.1.2 Classi.....                                              | 6        |
| 6.1.3 Esempi di proprietà modificabili e relativi valori. .... | 9        |
| 6.2 DIVISIONS E SPANS .....                                    | 11       |
| 6.2.1 Layers e proprietà Z-index.....                          | 12       |
| 6.3 ELEMENTO HEAD .....                                        | 14       |
| 6.3.1 Il tag <i>&lt;meta&gt;</i> .....                         | 15       |

## 6. HTML (continua)

### 6.1 Fogli di stile (CSS- Cascading StyleSheet)

Per stile si intende un insieme di istruzioni di formattazione. Un foglio di stile CSS definisce quindi gli stili che possono essere applicati a pagine o elementi delle pagine. I fogli di stile sono una specifica HTML sviluppata dal World Wide Web Consortium (W3C) e contribuiscono a tenere separato *il contenuto dalla formattazione*. La modalità più flessibile per l'utilizzo di uno stile è quella di definire lo stile in file esterni (stylesheet), che hanno estensione css (NomeFileStile.css). In questo modo potrà essere utilizzato da altri documenti, in qualunque numero e appartenenti a siti diversi, includendo il foglio di stile mediante un indirizzo di tipo URL. Includendo gli elementi di stile in documenti separati si possono ottenere documenti “portabili” ai quali possono essere applicati differenti stili in rapporto al mezzo di visualizzazione delle pagine. Altre possibilità di applicare gli stili consistono nella creazione di stili a livello dell'intero documento o a livello di specifici tag del documento. Nel primo caso lo stile è definito **all'interno dell'intestazione del documento (tag head)**, nel secondo ricorrendo all'attributo *style* per i singoli tag. Queste modalità sono anche dette **stili in linea**.

E' importante considerare che il comportamento degli stili è “**a cascata**”. Poiché in un documento possono essere utilizzati più stili, il funzionamento a cascata significa che un secondo stile definito per un blocco di documento si sovrappone al primo (verrà quindi utilizzato il secondo per la visualizzazione della pagina), se il secondo stile si riferisce ad una “parte” del documento già trattata dal primo. Lo stesso si verifica per un terzo stile rispetto al secondo e così di seguito. Se ad es. si usa uno stile per una parte specifica del documento (es.: intestazione h3) per quella parte verrà usato lo stile lì definito, sovrapponendosi all'eventuale stile definito per quella parte da un foglio esterno. In altri termini si applica lo stile “più vicino” alla parte di documento.

La possibilità di usare più stili, unitamente alla caratteristica del funzionamento a cascata, rende possibile, ad es., la creazione di un determinato stile per tutto il sito con **variazioni specifiche solo per alcune sezioni**, contribuendo alla creazione di un “look and feel” coerente dell'intero sito. Il corretto funzionamento dei fogli di stile richiede che i tag siano privi di errori. Nel caso di browser che non supportano l'uso di stylesheet i documenti saranno formattati come se non si fosse fatto uso dello stylesheet.

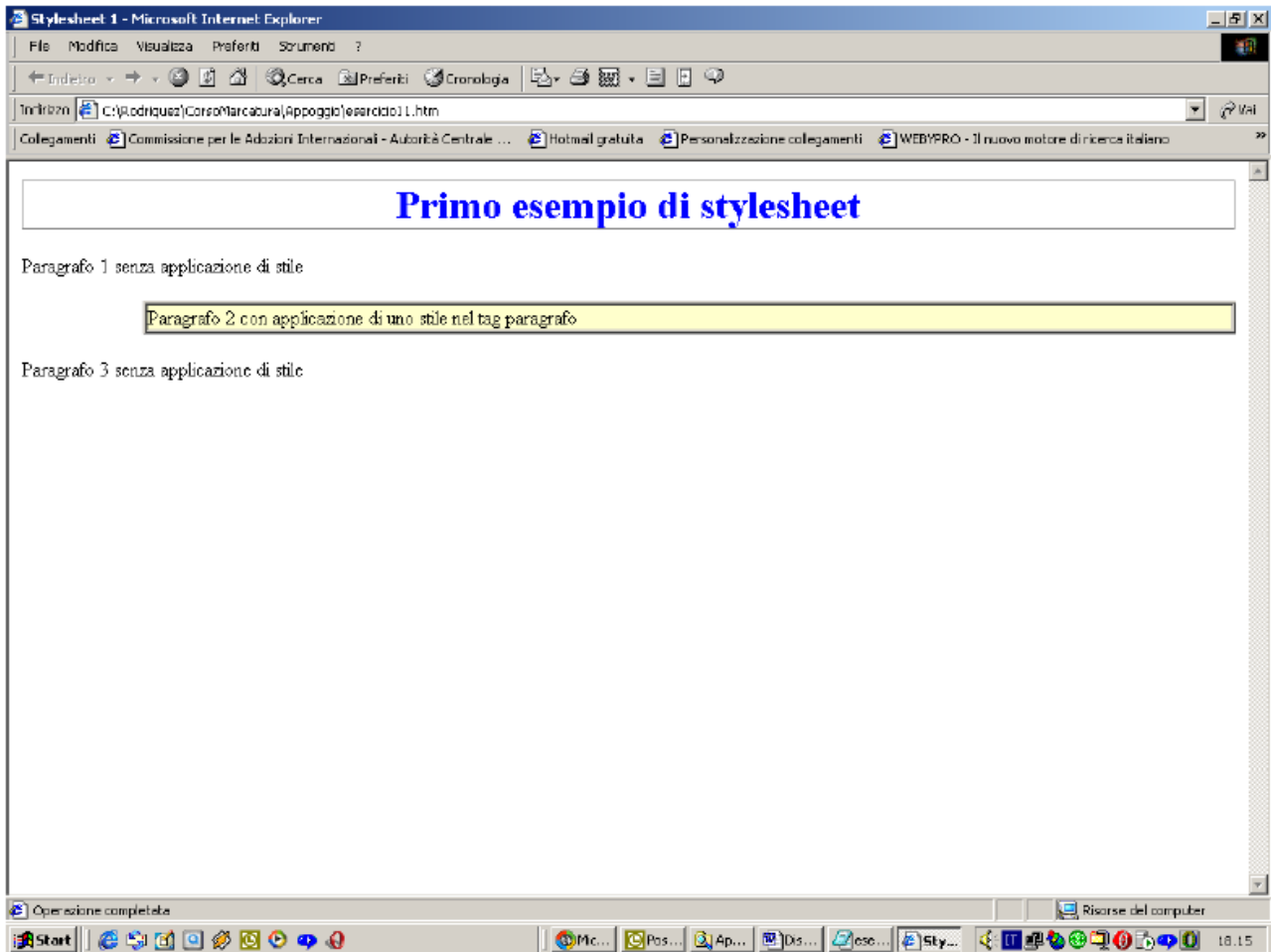
## Esempio sintassi

```
body {
    color: black;
    background:white;
}
strong {color: red}
```

Le istruzioni usate nei fogli di stile sono dette **regole di stile**. Le regole sono scritte indicando il nome del tag seguito dall'elenco delle proprietà dello stile incluse tra parentesi graffe. Ogni proprietà di stile è individuato con il nome della proprietà, i due punti e il valore da attribuire alla proprietà. Nel caso di più proprietà le stesse sono separate dal carattere punto e virgola. L'istruzione dell'esempio di sintassi precedente imposta i caratteri in nero e lo sfondo in bianco, mentre le parti di testo enfatizzate con strong appariranno in rosso.

## Esercizio:

```
<html>
  <head>
    <title>Stylesheet 1</title>
    <style>
      h1{
        color: blue;
        border: thin groove;
        text-align: center;
      }
    </style>
  </head>
  <body>
    <h1>Primo esempio di stylesheet</h1>
    <p> Paragrafo 1 senza applicazione di stile</p>
    <P style="margin-left: 10%; border: ridge; background: #ffffcc">
      Paragrafo 2 con applicazione di uno stile nel tag paragrafo</p>
    <P>Paragrafo 3 senza applicazione di stile</p>
  </body>
</html>
```



Si tenga presente che in fase di stampa di una pagina web può accadere che il browser non tenga conto del colore di sfondo definito e consideri solo il colore dei caratteri del testo. Per questo motivo impostare il carattere del testo in bianco potrebbe generare in stampa una pagina vuota.

### Esempio foglio di stile esterno

Stile.css

```
/* stile per intestazioni di livello 1 */
h1{color: blue; border: thin groove;text-align: center;}
/* stile per i paragrafi */
p {color: red; border: ridge; background: #ffffcc;}
```

```
<html>
  <head>
    <title>Esempio stylesheet esterno</title>
    <link href="Stile.css" rel="stylesheet" type="text/css">
  </head>
<body>
```

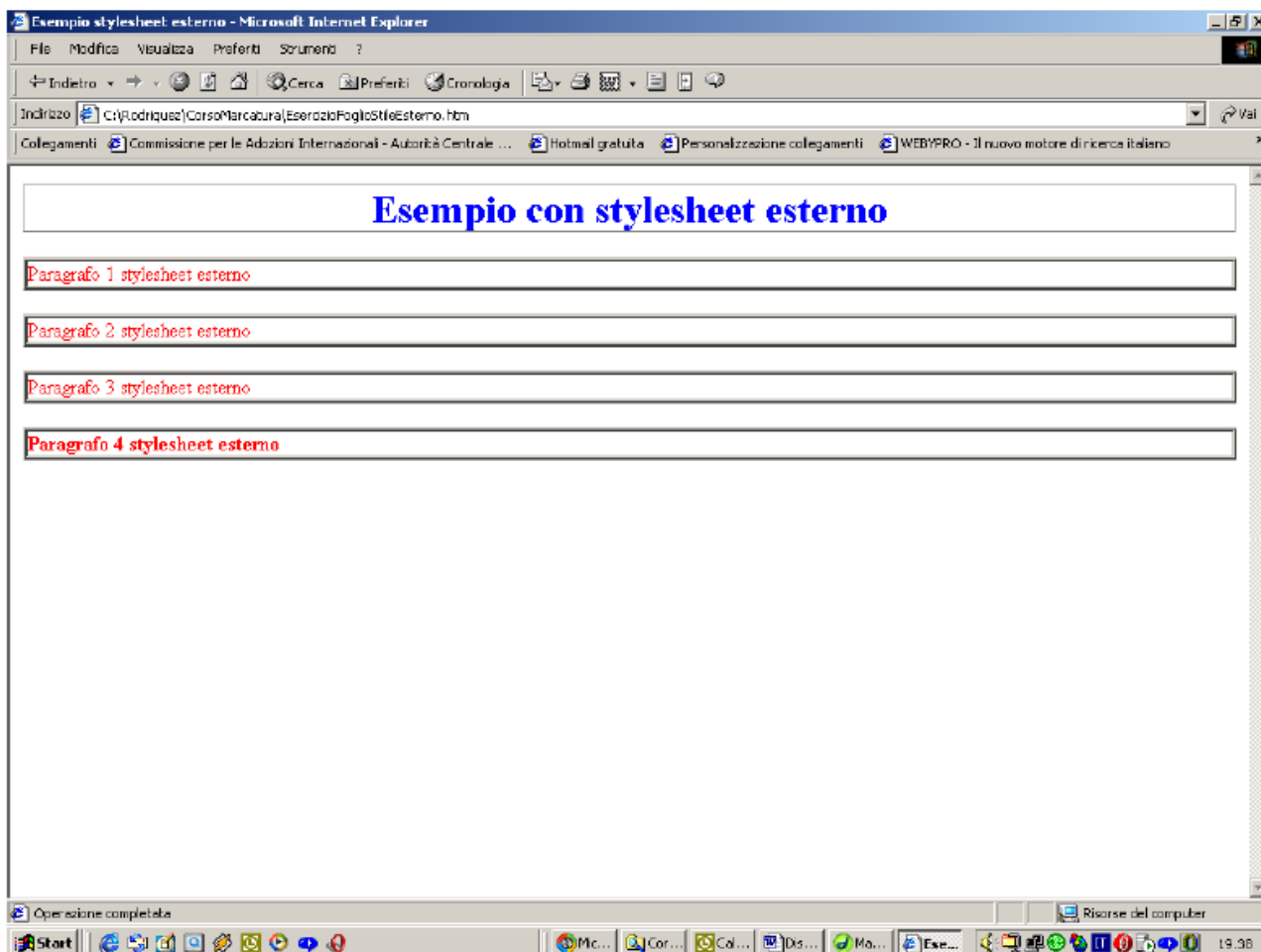
```

<h1>Esempio con stylesheet esterno</h1>
<p> Paragrafo 1 stylesheet esterno</p>
<p> Paragrafo 2 stylesheet esterno</p>
<p> Paragrafo 3 stylesheet esterno</p>
<p> <b> Paragrafo 4 stylesheet esterno</b></p>
</body>
</html>

```

Nel codice è presente l’istruzione che fa riferimento ad un CSS esterno. L’attributo *rel* impostato al valore *stylesheet* comunica al browser che l’attributo href si riferisce ad un foglio di stile. Il link al foglio di stile dovrebbe essere inserito nell’elemento head del documento HTML. Si noti che se una pagina è collegata a un foglio di stile esterno, gli stili incorporati o in linea creati per quella pagina **estenderanno o avranno la priorità** sulle proprietà specificate nel foglio di stile esterno.

Il codice genera il seguente risultato:



### 6.1.1 Regole dello stile

Ciascuna regola di stile (definizione dello stile), è composta da un **selettore** seguito dalle relative **proprietà** e **valori**. Il selettore è usato per creare il link tra la regola e il tag html.

*selettore {proprietà: valore; proprietà: valore; ... }*

E' consigliabile usare il punto e virgola anche dopo l'ultimo valore.

Esempi di regole di stile definite in un foglio di stile:

```
H1 { font-size: small; color: red;}
H2 { font-size: large; color: blue;}
```

Nell'esempio H1 e H2 sono i selettori che contengono le proprietà di formattazione dei tag HTML standard da modificare con l'attribuzione ai tag dei valori specificati per ciascuna proprietà (fontsize è una proprietà e small è il valore della proprietà font-size). Possono essere specificate più proprietà per un unico selettore separandole con un punto e virgola.

### 6.1.2 Classi

Le **classi di stile** sono un insieme di istruzioni di formattazione che possono essere applicate ad una pluralità di tag HTML. L'applicazione delle classi avviene aggiungendo esplicitamente la clausola *class* al tag di apertura dell'elemento HTML al quale applicarla e pertanto le classi non vengono applicate in modo automatico per ogni ricorrenza del tag. L'uso delle classi apporta ulteriore flessibilità nella costruzione della pagine. E' possibile definire gruppi di stile differenti da applicare a tag uguali, il che significa che le classi consentono di alterare solo alcuni elementi. Le classi possono essere definite sia a livello documento sia nei *css* esterni.

Sintassi

*Selettore.NomeDellaClasse {proprietà: valore; proprietà: valore; . . . . }*

La sintassi prevede l'indicazione del nome del tag seguito da un punto e dal (i) valore(i) degli attributi specificati per quella classe. Il punto non deve essere preceduto e seguito da spazi. Allo stile si attribuisce un nome che viene utilizzato nel tag mediante la clausola *class* seguita dal nome attribuito. Un'altra caratteristica delle classi è quella di poter essere **anonime**. Ciò significa che l'insieme degli elementi da alterare, previsti dallo stile, possono essere usati per alterare qualsiasi

elemento (non si riferiscono ad un elemento particolare). In questo caso la sintassi non prevede l'indicazione di un tag ma semplicemente di un punto seguito dal nome della classe.

### Esempi di sintassi

```
h3 {  
    Color: blue;  
    border: thin groove;  
}
```

Questo stile è applicato a tutti gli elementi H3 del documento

```
h3.ScegliElementoh3{  
Color: red;  
Margin-left: 40%;  
}
```

Questo stile è applicato solo agli elementi h3 che lo richiamano esplicitamente

```
.ClasseAnonima{  
text-align: right;  
color: green;  
}
```

Questo stile può essere applicato a qualsiasi tipo di tag, richiamandolo esplicitamente, come di seguito illustrato:

```
<h3 class="ClasseAnonima"> ... </h3>  
<p class="ClasseAnonima"> ... </p>
```

Si noti che l'elemento h3 è già stato alterato secondo la formattazione prevista dalla classe "ScegliElemento" ed è possibile, ove necessario, applicare un'altra formattazione.

## Esercizio

```
<html>
  <head>
    <title>Altro esempio di stylesheet</title>
    <style>
      h1{
        color: red;
        border: thin groove;
      }
      h2 {
        color: green;
        margin-left: 60%;
      }
      .myid {
        text-align: right;
        color: purple;
      }
      .myid2 {
        text-align: right;
      }
    </style>
  </head>
  <body>
    <h1>Altro esempio di stylesheet</h1>
    <p> Paragrafo senza applicazione di stile.</p>
    <P style="margin-left: 10%; border: ridge; background: #ffffcc">
      Paragrafo con applicazione di uno stile nel tag paragrafo (passato
      come attributo).</p>
    <P> Altro paragrafo senza applicazione di stile</p>
    <h2> Intestazione di tipo h2</h2>
    <p class="myid"> Questo è il testo che segue l'intestazione h2.</p>
    <h3 class="myid"> Altra intestazione (tipo h3)</h3>
    <p class="myid2">Etichetta</p>
    <p style="margin-left: 30%; color: blue">
      Altro testo collocato al centro dello schermo.</p>
  </body>
</html>
```

Il codice produce il seguente risultato:



### 6.1.3 Esempi di proprietà modificabili e relativi valori.

#### Caratteri

font-style: normal|italic

font-weight: normal|bold|bolder|lighter

bolder e lighter sono misure relative

font-size: [small|medium|large| | [smaller|larger]

smaller e larger sono misure relative

#### Testo

text-transformation: none|capitalize|uppercase|lowercase

il valore capitalize trasforma le iniziali in caratteri maiuscoli

text-align: left|right|center|justify

text-indentation: length|percentage

con il valore percentage il rapporto tra l'indentazione ed il resto del testo è mantenuto nel caso in cui la finestra del browser è ridimensionata.

```
<style type="text/css">
body {margin-left: 10%; margin-right: 10%; }
</style>
```

Questa istruzione imposta i margini della pagina web al 10% dell'ampiezza totale della finestra del browser. Trattandosi di una misura relativa le proporzioni dei margini non varieranno nel caso di ridimensionamento manuale della finestra del browser.

### Indentazione

```
P {text-indent: 3em; margin-top: 0; margin-bottom:0;}
```

Con questa istruzione la prima riga di ogni paragrafo rientra di 2 em e la spaziatura tra paragrafi è azzerata.

### Dimensione dei caratteri

```
H1 {font-size: 150%;}
H2 {font-size: 100%;}
H3 {font-size: 200%;}
```

Queste istruzioni impostano le dimensioni dei primi tre livelli di intestazione con valori espressi in percentuale rispetto alle dimensioni del testo normale

### Riquadri

```
Margin: length|percentage|auto {1,2,4}
Border-width: thin|thick|medium|length {1,2,4}
Padding: length|percentage {1,2,4}
```

Nel caso dei margini può essere opportuno usare i valori assoluti. Nel caso di valori percentuali e di ridimensionamento della finestra i riquadri potrebbero sovrapporsi in modo disordinato.

```
Border-color: Valore {1,2,4}
Border-style: none|dotted|dashed|solid|double|groove|ridge {1,2,4}
Width: length|percentage|auto
```

E' consigliata l'indicazione del valore in percentuale

```
Height: length|auto
```

E' consigliata l'indicazione del valore assoluto

La notazione {1,2,4} indica che possono essere specificati 1,2 o 4 valori che verranno applicati con la modalità riportata di seguito (con riferimento alla proprietà margin):

| <i>Numero valori indicati</i> | <i>Ordine di applicazione</i>                                                                |
|-------------------------------|----------------------------------------------------------------------------------------------|
| <i>4</i>                      | Con la sequenza margine alto, destro, fondo, sinistro                                        |
| <i>2</i>                      | Il primo valore è applicato in alto e in fondo, il secondo è applicato a sinistra e a destra |
| <i>1</i>                      | Lo stesso valore è applicato ai 4 margini                                                    |

## 6.2 Divisions e spans

L'applicazione degli stili può essere eseguita anche su **sezioni di testo** tramite i tag *div* e *span*.

*Div* suddivide il testo analogamente a come si suddivide un articolo in titolo, sottotitolo, paragrafi, ecc.

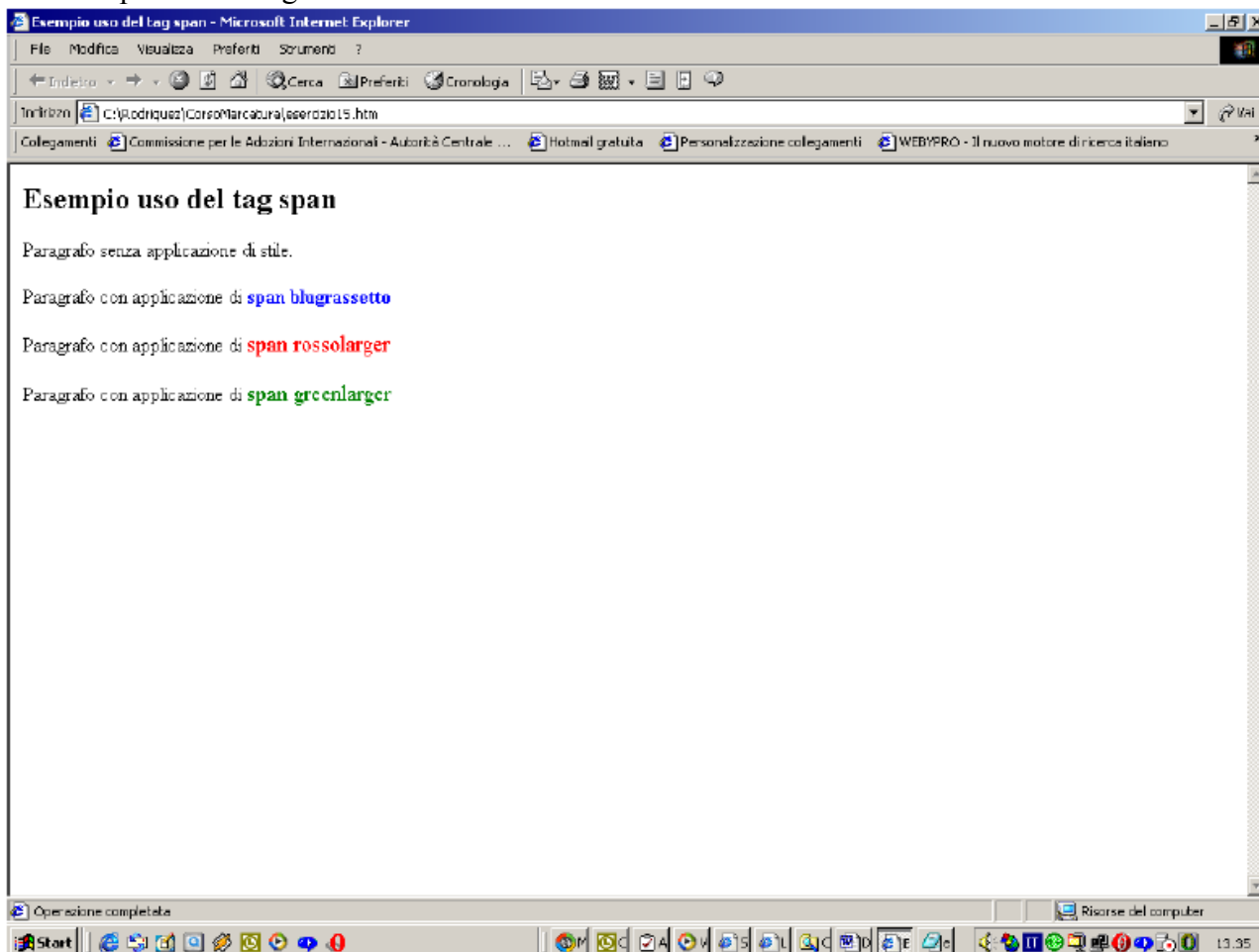
*Span* è usato per selezionare una parte specifica di testo (ad es. una parola o una serie di parole).

In entrambi i casi si aggiunge ai tag di apertura la clausola *class*.

### Esercizio

```
<html>
  <head>
    <title>Esempio uso del tag span</title>
    <style>
      .blugrassetto {
        color: blue;
        font-weight:bold;
      }
      .rossolarger{
        color: red;
        font-size:larger;
      }
      .greenlarger{
        color: green;
        font-size: larger;
      }
    </style>
  </head>
<body>
  <h2>Esempio uso del tag span</h2>
  <p> Paragrafo senza applicazione di stile.</p>
  <p>Paragrafo con applicazione di <span class="blugrassetto">span
    blugrassetto</span></p>
  <p>Paragrafo con applicazione di <span class="rossolarger">span
    rossolarger</span></p>
  <p>Paragrafo con applicazione di <span class="greenlarger">span
    greenlarger</span></p>
</body>
</html>
```

Il codice produce il seguente risultato:



## 6.2.1 Layers e proprietà Z-index

Una pagina può essere strutturata disponendo elementi su strati collocati su diversi livelli. Per sovrapporre gli elementi occorre che gli stessi siano posizionati sulla pagina mediante i valori top, left, bottom, right e successivamente si usa la proprietà z-index per indicare **l'ordine di sovrapposizione**.

### Sintassi

z-index: n

Il valore numerico  $n$  indica il livello dello strato. A numeri maggiori corrispondono livelli più alti e quindi un elemento con livello 2 sarà posizionato “sopra” un elemento con livello 1. Se il valore di z-index non è specificato gli elementi saranno sovrapposti secondo l'ordine con il quale appaiono nel codice HTML. Lo stesso si verifica se due elementi hanno lo stesso valore di z-index.

Position: absolute|relative

Le divisioni devono essere posizionate sullo schermo in modo che l'angolo in alto a sinistra parta dai pixel 0,0

Left: n

Top: n

Indica il posizionamento della divisione in pixel

Width: n

Height: n

Indica la dimensione della divisione in pixel

### Esercizio

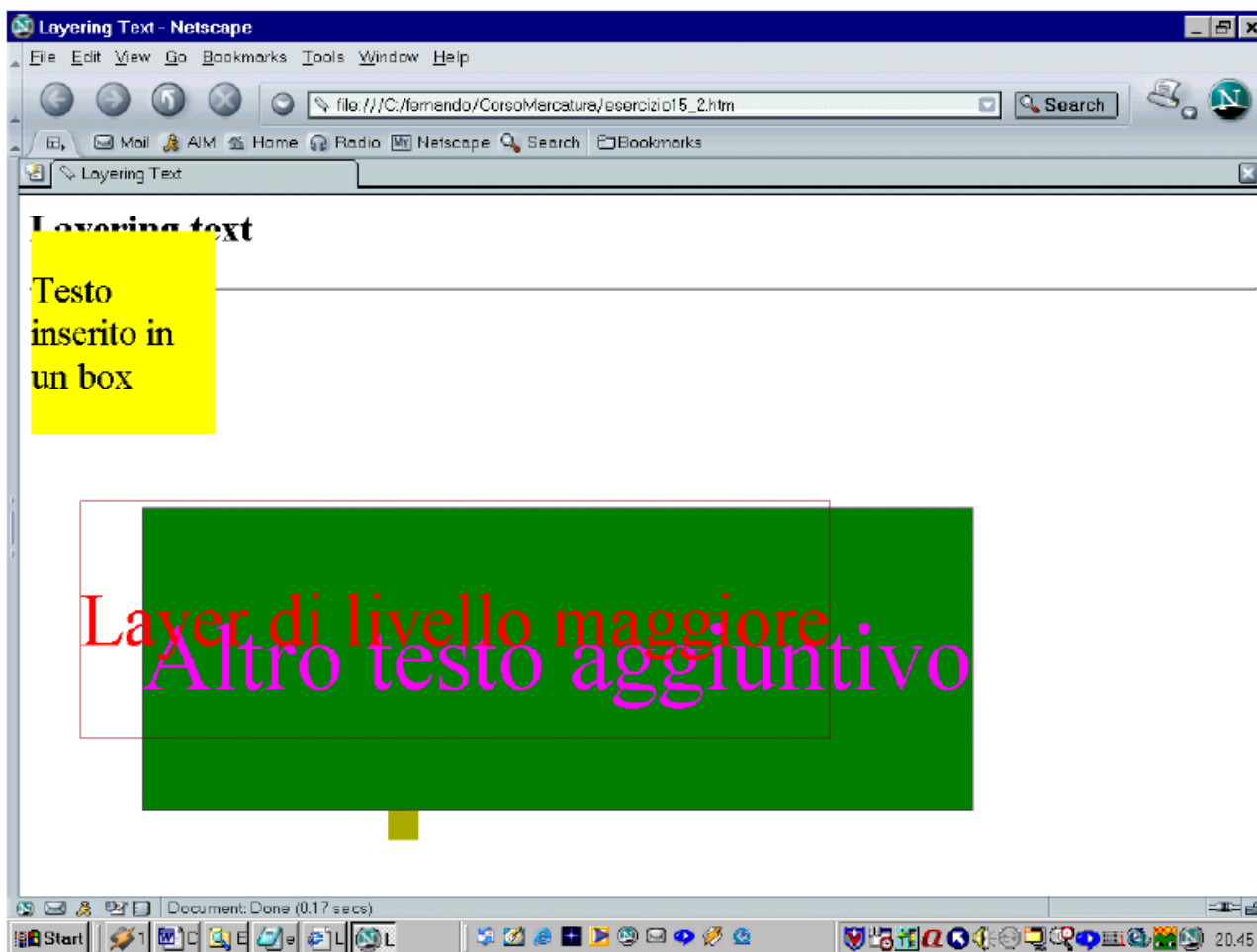
```
<html>
  <head>
    <title>Layering Text</title>
  </head>
  <body>
    <h1> Layering text</h1>
    <div style="z-index: 2; left: 50px; top: 250px;
      position: absolute; color: red; text: white;
      font-size: 36pt; border: thin groove;">
    <p>Layer di livello maggiore</p>
    </div>

    <div style="z-index: 1; left: 100px; top: 255px;
      position: absolute; font-size: 46pt; color: magenta;
      background-color: green; border: thin groove">
    <P>Altro testo aggiuntivo</p>
    </div>

    <div style="z-index: 1; left: 10; top: 30px; width: 150px;
      position: absolute; background-color: yellow;
      color: black; font-size: 18pt;">
    <p>Testo inserito in un box
    </div>

    <div style="position: absolute; top: 280px; left: 300px;
      width: 25; background-color: #aeae00; color: blue;
      font-size: 16pt; font-style: italic; z-index=2">
    <p> Altro testo in basso a destra
    </div>
    <hr>
  </body>
</html>
```

Il codice produce il seguente risultato



### 6.3 Elemento head

#### *Document Type Declaration*

Per un documento HTML la dichiarazione del tipo di documento, in genere, è:

```
<!doctype html public "-//w3c//dtd html 4.0//en"  
"http://www.w3.org/TR/PR-html4.0/loose.dtd" >
```

4.0 indica il numero di versione di HTML, en indica la lingua inglese

Per l'HTML il W3C ha creato i seguenti DTDs:

- loose
- strict
- transitional
- frameset

Le DTDs contengono le regole da rispettare se un documento deve essere conforme agli standard HTML (*raccomandazioni*). La conformità delle pagine può essere verificata con appositi pacchetti software (*validatori*).

### 6.3.1 Il tag <meta>

Il tag head del documento può contenere, oltre al titolo, anche **informazioni di controllo** usate dai server e dai browser come ad es. una lista di parole chiave file da includere nel documento, ecc. nonché per l'indicizzazione da parte dei motori di ricerca. Si tenga comunque presente che alcuni motori di ricerca ignorano i tag meta ed indicizzano le parole che si trovano direttamente nel corpo della pagina.

Le informazioni si aggiungono tramite il tag “meta” i cui attributi più comuni sono “name” e “content”. Per l'attributo name non esistono valori predefiniti ed i più usati sono “keywords” e “description”. In genere, inserendo le parole chiave, si includono anche le varianti ortografiche per quelle parole che risultano più spesso scritte in modo errato.

Un valore di attributo per i tag meta è “http-equiv” che invia informazioni aggiuntive al browser. Se si imposta l'attributo http-equiv uguale a “refresh” unitamente ad un intervallo di tempo e si imposta un URL tramite l'attributo “content” dopo il periodo di tempo specificato la pagina viene ricaricata automaticamente o il browser viene reindirizzato al nuovo URL indicato. Le informazioni del tag meta non modificano l'aspetto della pagina Web.

Esempio

```
<!doctype html public "-//w3c//dtd html 4.0//en"
"http://www.w3.org/TR/PR-html4.0/loose.dtd" >
<html>
  <head>
    <base href="http://www.governo.it">
    <link rel="Stylesheet" href=".test.css" type"text/css"
      media="screen">
    <meta name="author" content="Anonimo del 700">
    <meta http-equiv="expires" content="Data + ora di scadenza">
    <meta name="description" content="Catalogo della Società Pizza e
      Mandolini.">
    <title> Pizza e Mandolini S.p.A. Catalogo in linea</title>
  </head>
  <body>
  </body>
</html>
```

Nell'esempio il primo meta tag si riferisce all'autore del documento.

Il secondo, contenente una data di scadenza, comunica al browser di **caricare nuovamente** la pagina dal server dopo la data di scadenza indicata, invece di utilizzare l'eventuale pagina memorizzata nella memoria cache del computer. Si noti che, comunque, il browser può essere configurato in modo da stabilire le modalità di (ri)caricamento delle pagine.

Il terzo meta tag contiene la descrizione del documento che dovrà essere usata dai motori di ricerca.