

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;
}

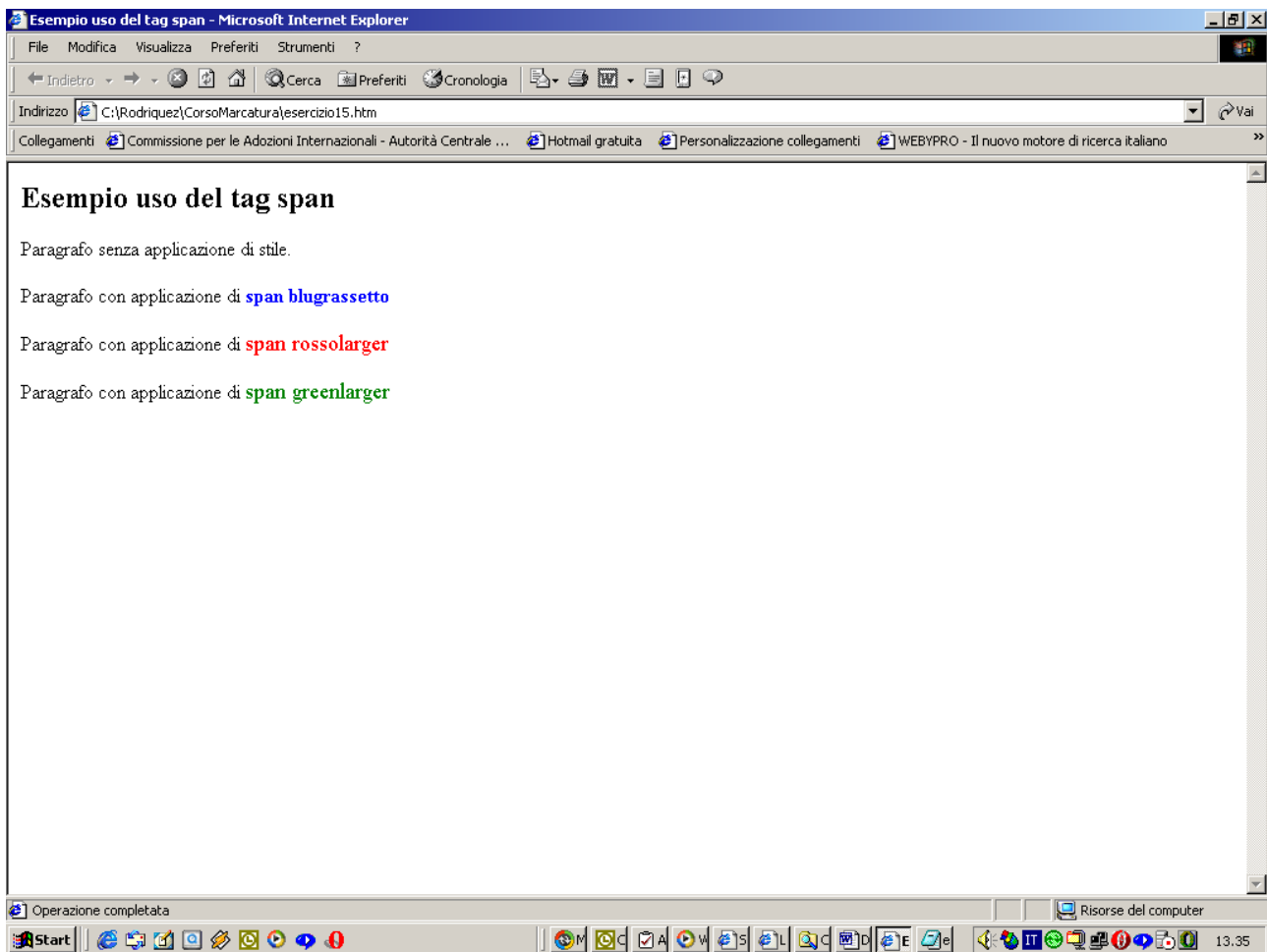
-->

</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:



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

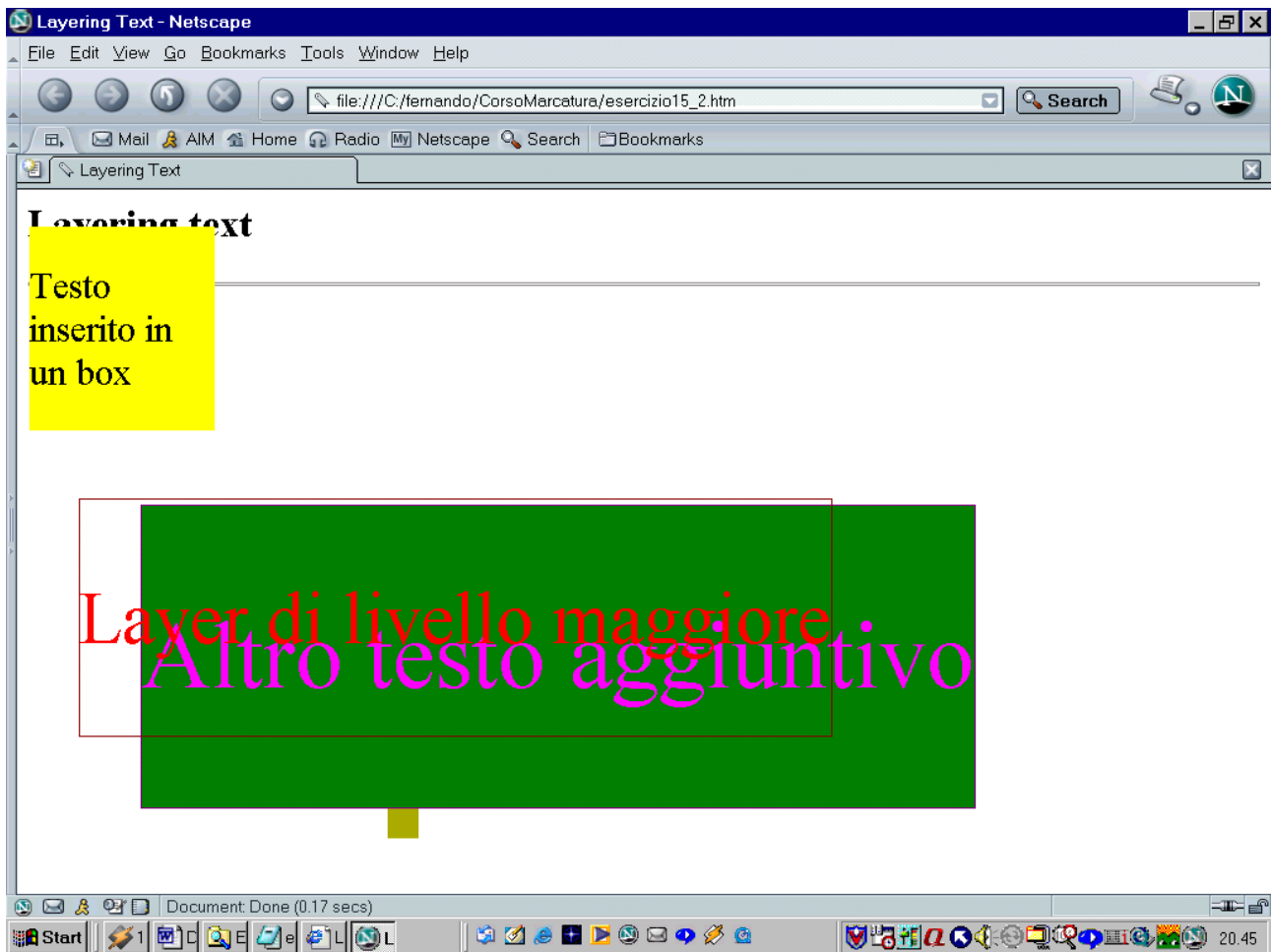
    </div>

    <hr>

  </body>

</html>
```

Il codice produce il seguente risultato



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*).

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>
```

```
<meta name|http-equiv="StringaCaratteri" content="StringaCaratteri">
```

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.

JavaScript (esercizi)

Gli script possono essere inclusi nelle singole pagine HTML o in file separati, con la conseguente possibilità di un loro riuso.

Javascript aggiunge dinamicità alle pagine web ed è usato, ad es., per animare testi e immagini e per validare l'input dell'utente.

I JavaScript sono interpretati dall'Application Server quando il web client richiede un documento HTML contenente lo script. I risultati dell'interpretazione saranno poi trasferiti al web server.

- gli script possono essere inclusi in singoli documenti HTML o in file separati con possibilità di riuso
- le variabili e le istruzioni sono case-sensitive
- non è necessario dichiarare le variabili
- le variabili non hanno un tipo di dato definito; l'assegnazione avviene dinamicamente nel momento in cui è valutata l'espressione della variabile
- negli script possono essere usati i seguenti tipi di dato
 - numerici
 - logici (booleani)
 - stringhe
 - null
 - funzioni
 - oggetti (metodi e proprietà)
- l'utente può creare propri oggetti e definire i loro metodi e le loro proprietà (sono disponibili anche oggetti *“predefiniti”*)

Esercizio

```
<html>
<title>Alert Javascript</title>
<body>
<h1 align="center">Un esempio originale</h1>
<script language= "JavaScript" >
<!--

// Commento: una finestra di dialogo

var hello = "Hello world!";
alert (hello);

//-->

</script>
</body>
</html>
```

Il codice produce il seguente risultato:

