

# Metodologie Informatiche Applicate al Turismo

## 10. Cascading Style Sheets (CSS)

Paolo Milazzo

Dipartimento di Informatica, Università di Pisa

<http://www.di.unipi.it/~milazzo>

[milazzo@di.unipi.it](mailto:milazzo@di.unipi.it)

Corso di Laurea in Scienze del Turismo

A.A. 2013/2014

# Sommario

- 1 Introduzione ai CSS
- 2 CSS e HTML
  - Usare CSS in HTML
- 3 La sintassi di CSS
  - Definizione
  - I tag `<div>` e `<span>`
  - Alcune proprietà di base
- 4 Il modello visuale di CSS
  - Il box model
  - Il posizionamento degli elementi

# Introduzione ai CSS (1)

- HTML è nato con l'idea di descrivere i contenuti in maniera separata rispetto agli aspetti di presentazione
- Le prime versioni dei browser permettevano agli **utenti** di definire l'aspetto dei vari elementi dei documenti HTML visualizzati (dimensione dei font, colori, ecc...)
- Successivamente la diffusione e l'evoluzione dei siti web ha reso sempre più importante la **gestione centralizzata** degli aspetti di presentazione
- La possibilità di visualizzare documenti HTML in dispositivi di natura diversa (PC, smartphones, ecc...) ha stimolato ulteriormente la separazione tra gli aspetti di contenuto e quelli di presentazione, promuovendo l'utilizzo di **linguaggi di stile** specializzati
- **Cascading Style Sheets (CSS)** è il linguaggio di stile che, tra le numerose proposte, si è imposto come “compagno” standard di HTML

## Introduzione ai CSS (2)

Esistono diversi **livelli** di specifica di CSS:

- CSS level 1 (1991): è un linguaggio di formattazione visiva abbastanza semplice. Permette di specificare caratteristiche tipografiche e di presentazione di ogni elemento di un documento HTML
- CSS level 2 (1998): attualmente in versione 2.1, è un linguaggio di layout sofisticato e complesso
- CSS level 3 è stata sviluppata negli ultimi anni: migliora/estende la gestione di layout sofisticati, colori, sfondi, bordi, animazioni, trasformazioni, etc...

Il supporto dei browser a CSS è complesso e difficile da realizzare. Attualmente, comunque, le versioni più recenti dei principali browser supportano completamente la versione 2.1 e iniziano a supportare abbastanza bene la versione 3.

# Usare CSS in HTML (1)

- CSS fornisce informazioni sull'aspetto di elementi di un documento HTML descritti tramite i tag
- Si basa su regole simili alla seguente:

```
p { font-family: Arial; font-size: 12pt; }
```

che stabilisce che tutti per visualizzare il contenuto di tutti i tag <p> nel documento deve essere usato font Arial di dimensione 12 punti

## Usare CSS in HTML (2)

- HTML prevede l'uso di CSS in (almeno) tre modi diversi:
  - ▶ Posizionato direttamente nel tag di riferimento tramite l'attributo `style`
  - ▶ All'interno di un tag `<style>`
  - ▶ Importato da un file esterno usando il tag `<link>`
- L'ultimo modo, che prevedono di importare un documento CSS (detto **foglio di stile**) da un file esterno, consente di scrivere il codice CSS una volta sola per tutte le pagine di un sito
  - ▶ basta importare lo stesso file in tutti documenti HTML del sito
- Vediamo degli esempi....

## Usare CSS in HTML (3)

CSS posizionato direttamente nel tag di riferimento tramite l'attributo `style`:

```
<html>
  <head>
    <title>Pagina di prova</title>
  </head>
  <body>
    <h1 style="color:blue;">Questo titolo deve essere blu</h1>
    <p>Contenuto della pagina di prova</p>
  </body>
</html>
```

- In questo caso la regola CSS sarebbe

```
h1 { color: blue; }
```

ma nell'attributo `style` si indica solo il contenuto tra parentesi

- la regola CSS si applica solo a quello specifico tag `<h1>`, non a tutti gli altri

# Usare CSS in HTML (4)

CSS all'interno di un tag <style>:

```
<html>
  <head>
    <title>Pagina di prova</title>
    <style type="text/css">
      h1 { color: blue; }
    </style>
  </head>
  <body>
    <h1>Questo titolo deve essere blu</h1>
    <p>Contenuto della pagina di prova</p>
  </body>
</html>
```

- Stesso risultato del precedente, ma in questo caso se ci fossero altri tag <h1> anche il loro contenuto diventerebbe blu.

# Usare CSS in HTML (5)

CSS indicato dal tag `<link>`:

```
<html>
  <head>
    <title>Pagina di prova</title>
    <link type="text/css" rel="stylesheet" href="extfile.css">
  </head>
  <body>
    <h1>Questo titolo deve essere blu</h1>
    <p>Contenuto della pagina di prova</p>
  </body>
</html>
```

dove `extfile.css` è il seguente

```
h1 { color: blue; }
```

- Stesso risultato del precedente, tutti i titoli `<h1>` diventano blu.
- Il file `extfile.css` può essere “linkato” anche dalle altre pagine dello stesso sito (per uniformare l’aspetto)

# Gli attributi id e class

- HTML prevede che tutti i tag di un documento possano avere attributi `id` e `class`
  - ▶ `id` deve essere associato ad un identificativo univoco di un'occorrenza di un tag nell'intero documento (non può essere ripetuto)
  - ▶ `class` deve essere associato ad un identificativo di classe che può essere usato più volte nel documento
- CSS può usare i valori di `id` e `class` per specificare che una regola si applica ad uno specifico elemento del documento o a tutti gli elementi della stessa classe (vedremo...)

# La sintassi di CSS (1)

- Un foglio di stile CSS è costituito da una sequenza di **regole**
- Ogni regola consiste di un **selettore** e di una o più **dichiarazioni** racchiuse tra parentesi graffe (e ogniuna terminata da “;”)
- Ogni dichiarazione consiste di una **proprietà** ed un **valore** (separati da “:”)

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

- Il selettore è solitamente il tag di cui la regola specifica l'aspetto
- Le proprietà sono caratteristiche di stile che possono essere associate agli elementi. Sono definite dalla specifica di CSS e sono in tutto 53 per CSS level 1 e ben 121 per CSS level 2
- I valori dipendono dalla proprietà a cui sono assegnati...

```
p { font-family:Arial; font-size:12pt; }
```

## La sintassi di CSS (2)

selettore { proprietà<sub>1</sub>:valore<sub>1</sub>; proprietà<sub>2</sub>:valore<sub>2</sub>; ... }

Il selettore può avere tante forme, tra cui:

- **Selettore di tipo.** Un tag (senza parentesi <,>): body, p, table, div, span, ...
- **Selettore di classe.** Può avere due forme E.val e E#val. Significano che la regola si applica a tutti gli elementi E in cui gli attributi class e id, rispettivamente, sono assegnati a val.

Inoltre ci sono **altri selettori** (che non vediamo) per selezionare un tag che si trova dentro un altro tag, la prima riga di un paragrafo, la prima lettera di un paragrafo, una parola su cui è posizionato il mouse, ecc...

# Un Esempio

```
<html>
<head>
  <title>Prova</title>
  <style type="text/css">
    li.pari { color:red; }
    li.dispari { color:blue; }
    p#chiusura { font-size:75%; }
  </style>
</head>
<body>
<p>Esempio di lista colorata:</p>
<ul>
  <li class="dispari">Primo</li>
  <li class="pari">Secondo</li>
  <li class="dispari">Terzo</li>
  <li class="pari">Quarto</li>
</ul>
<p id="chiusura">Una frase di
chiusura scritta in piccolo...</p>
</body>
</html>
```

Esempio di lista colorata:

- Primo
- Secondo
- Terzo
- Quarto

Una frase di chiusura scritta in piccolo...

## Il concetto di cascata

- Un concetto molto importante in CSS è il concetto di **cascata**: regole successive che si possono applicare ad un elemento del documento HTML devono essere applicate in sequenza
- Ad esempio, applicando in sequenza:
  - ▶ `p { font-family: Arial; font-size: 12pt; }`
  - ▶ `p { color: red; font-size: 11pt; }`
  - ▶ `p { margin-left: 15pt; color: green; }`otterremo lo stesso risultato di
  - ▶ `p { font-family: Arial; font-size: 11pt; margin-left: 15pt; color: green; }`
- Questa funzionalità è utile in particolare in due casi:
  - ▶ Quando si prevede di applicare **più fogli di stile** (usando più volte il tag `<link>`) allo stesso documento HTML
  - ▶ Quando si vogliono applicare **regole generali** (ad esempio a tutti i tag `<p>`) + **regole specifiche** per singoli elementi tramite l'attributo `style`

# I tag <div> e <span> (1)

- Con CSS si possono specificare regole di presentazione da associare a qualunque tag
- Talvolta, però, si vorrebbero definire regole da applicare
  - ▶ ad un'intera porzione del documento HTML (che potrebbero includere molti tag diversi),
    - ★ Ad esempio, per impostare il colore di sfondo di un'area del documento
  - ▶ o a un piccolo frammento di testo all'interno di un paragrafo (che potrebbe non essere delimitato da un tag)
    - ★ Ad esempio, per impostare il font di una parola in mezzo al testo
- Per rendere possibile la definizione di queste regole in HTML sono stati definiti i tag <div> e <span>

## I tag <div> e <span> (2)

- Il tag <div> viene usato per delimitare un'area (anche rilevante) del documento HTML a cui applicare delle regole stilistiche comuni;
- Esempi di utilizzo:
  - ▶ per suddividere il documento in sezioni, ognuna delimitata da un bordo colorato
  - ▶ per isolare l'area che contiene l'indice del sito in modo da posizionarla alla sinistra della pagina
- Il tag <span> viene usato per identificare un piccolo frammento (tipicamente di testo) all'interno di un paragrafo, una lista, una tabella, ecc...
- Esempi di utilizzo:
  - ▶ Per creare un'effetto evidenziatore su singole parole o porzioni di frase

## I tag `<div>` e `<span>` (3)

- Tecnicamente, la differenza tra `<div>` e `<span>` consiste nel fatto che:
  - ▶ `<div>` appartiene alla categoria degli elementi di tipo `block` (come i paragrafi, le tabelle, ecc...) , ovvero elementi che, se usati in sequenza, vengono visualizzati uno sotto l'altro con un po' di spazio tra uno e l'altro
  - ▶ `<span>` appartiene alla categoria degli elementi di tipo `inline` (come `<strong>`, `<b>`, ecc...), ovvero elementi che, se usati in sequenza, vengono visualizzati uno accanto all'altro
- Gli attributi `id` e `class` potranno poi essere usati per differenziare le varie occorrenze di `<div>` e `<span>` nel documento e applicarvi regole di stile diverse

## 1 tag <div> e <span> (4)

Facciamo un esempio di uso di <div> e <span>:

```
<html>
  <head><title>Prova div e span</title></head>

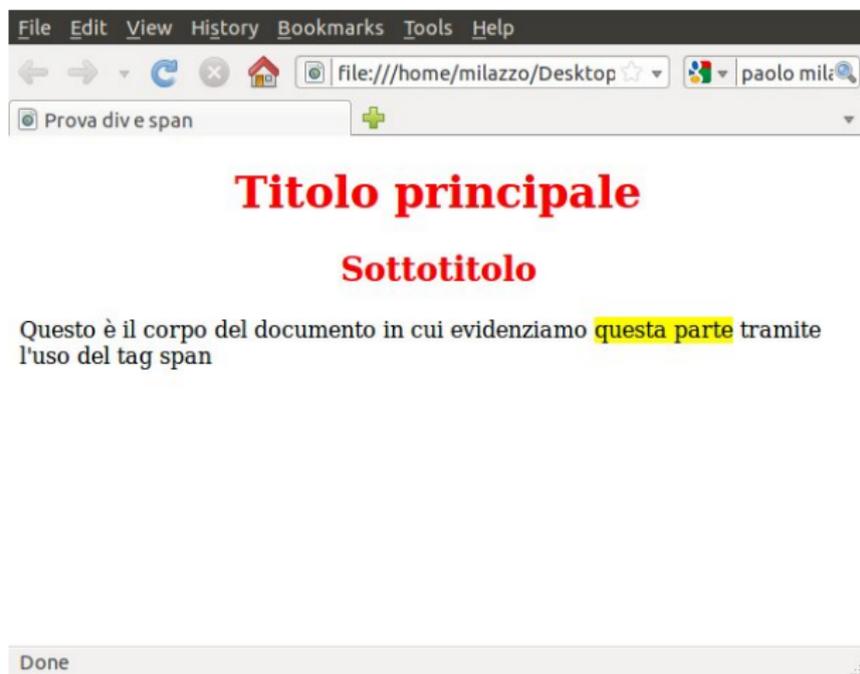
  <style type="text/css">
    div.titolo {
      text-align: center;
      color: red;
    }
    span#qp {
      background-color: yellow;
    }
  </style>

  <body>
    <div class="titolo">
      <h1>Titolo principale</h1>
      <h2>Sottotitolo</h2>
    </div>

    <div class="corpo">
      <p>Questo &grave; il corpo del documento in cui evidenziamo
        <span id="qp">questa parte</span> tramite l'uso del tag
        span</p>
    </div>
  </body>
</html>
```

## I tag <div> e <span> (5)

Il risultato dell'esempio:



## Un po' di esempi (1)

Per fare un po' di esempi consideriamo il seguente (body di) documento HTML

```
<body>
<h1>Ricetta: la <span class="piatto">pasta al pomodoro</span></h1>
<p>Vediamo ora come preparare un'ottima <span class="piatto">pasta
al pomodoro.</span> Innanzitutto, mettere a bollire in una
pentola capiente abbondante acqua.</p>
<p>Tagliare a fettine sottili la <span class="ingr">cipolla</span>
e soffriggerla con l'<span class="ingr">olio</span> in una
padella fino a quando non sar&agrave; ben dorata. In seguito
aggiungere i <span class="ingr">pomodori</span> e il
<span class="ingr">basilico</span>.</p>
<p>Quando bolle l'acqua, salarla e versare gli <span class="ingr">
spaghetti</span>. Una volta che gli spaghetti sono cotti,
scolarli e aggiungervi il sugo di pomodoro.</p>
</body>
```

## Un po' di esempi (2)

Vediamo come il browser visualizza il documento HTML:

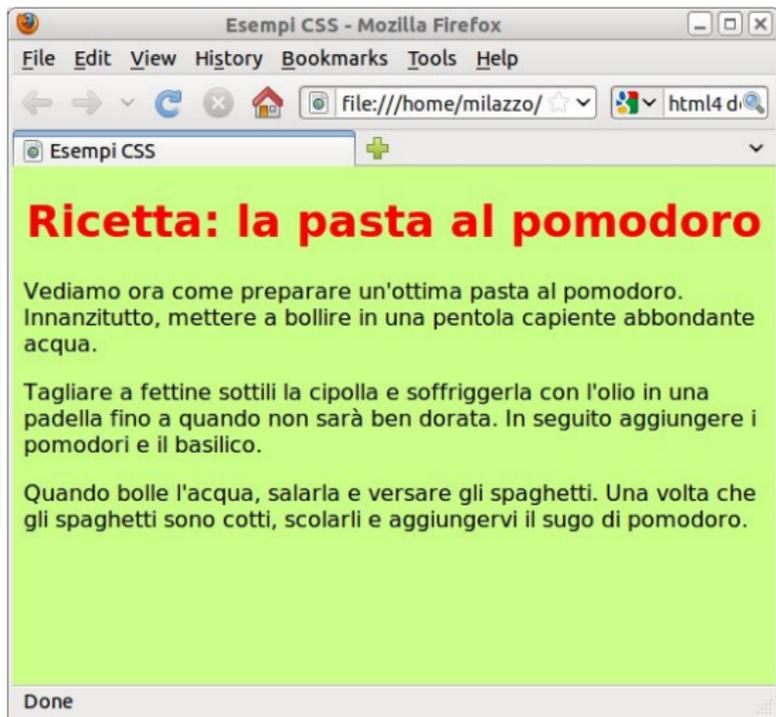


## Un po' di esempi (3)

Primo foglio di stile:

```
body
{
  background-color: #CCFF88;
  font-family: sans-serif;
}

h1
{
  color: red;
  text-align: center;
}
```



## Un po' di esempi (4)

Secondo foglio di stile (nota: `p:first-letter` è un selettore che indica la prima lettera di un paragrafo)

```
span.ingr
{
  background-color: yellow;
}
p:first-letter
{
  font-size: 300%;
  float: left;
}
```

Esempi CSS - Mozilla Firefox

File Edit View History Bookmarks Tools Help

file:///home/milazzo/ html4 d

Esempi CSS

## Ricetta: la pasta al pomodoro

Vediamo ora come preparare un'ottima pasta al pomodoro. Innanzitutto, mettere a bollire in una pentola capiente abbondante acqua.

Tagliare a fettine sottili la **cipolla** e soffriggerla con l'**olio** in una padella fino a quando non sarà ben dorata. In seguito aggiungere i **pomodori** e il **basilico**.

Quando bolle l'acqua, salarla e versare gli **spaghetti**. Una volta che gli spaghetti sono cotti, scolarli e aggiungervi il sugo di pomodoro.

Done

## Un po' di esempi (5)

Terzo foglio di stile (nota: `span:hover` è il selettore che indica il tag `span` su cui si è posizionato il mouse):

```
span.ingr, span.piatto
{
  background-color: yellow;
}
span:hover
{
  background-color: red;
}
p:first-letter
{
  font-size: 300%;
  float: left;
}
```

File Edit View History Bookmarks Tools Help

file:///home/milazzo/ html4 d

Esempi CSS

## Ricetta: la pasta al pomodoro

Vediamo ora come preparare un'ottima pasta al pomodoro. Innanzitutto, mettere a bollire in una pentola capiente abbondante acqua.

Tagliare a fettine sottili la cipolla e soffriggerla con l'olio in una padella fino a quando non sarà ben dorata. In seguito aggiungere i pomodori e il basilico.

Quando bolle l'acqua, salarla e versare gli spaghetti. Una volta che gli spaghetti sono cotti, scolarli e aggiungervi il sugo di pomodoro.

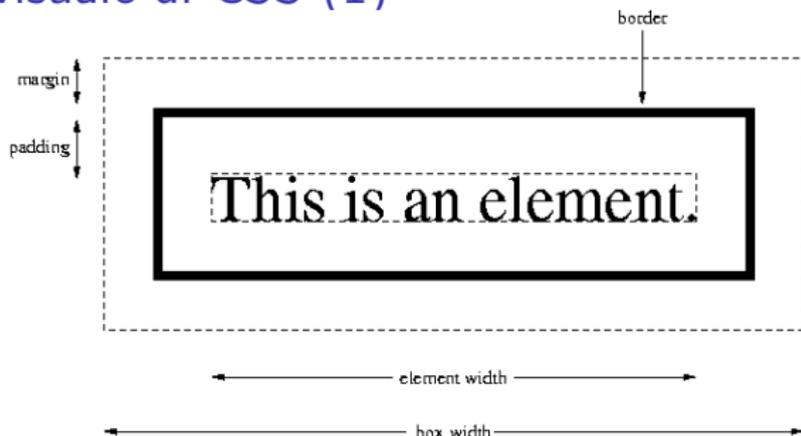
Done

## Alcune proprietà di base

Alcune (buona parte) delle proprietà che si possono usare nelle regole CSS sono illustrate **nell'appendice di questa lezione (slides aggiuntive sul sito del corso)**.

L'elenco completo delle proprietà si trova nella specifica di CSS definita dal consorzio W3C o in numerosi siti e tutorial sul web.

# Il modello visuale di CSS (1)



- Il modello visuale di CSS (ovvero l'insieme di principi riguardo la disposizione degli elementi) è detto **Box Model**
- Ogni elemento (tag) di un documento HTML è visto come una scatola che consiste di
  - ▶ Uno spazio di margine (`margin`)
  - ▶ Un bordo (`border`) che può essere trasparente, una linea continua o tratteggiata, ecc...
  - ▶ Uno spazio di separazione del contenuto dal bordo (`padding`)
  - ▶ Il contenuto vero e proprio del tag (`content`)

## Il modello visuale di CSS (2)

- Esistono numerose proprietà per specificare le caratteristiche di un tag secondo il box model
- Ad esempio, si possono impostare
  - ▶ larghezza e altezza dell'elemento
  - ▶ aspetto del bordo (tipo di linea, colore, spessore, ecc...)
  - ▶ dimensione del margine e del padding
  - ▶ colore di sfondo
- le proprietà si possono applicare anche a un singolo lato della “scatola” che rappresenta il tag
- la descrizione dei dettagli di queste proprietà può essere trovata **nell'appendice di questa lezione (slides aggiuntive sul sito del corso)**.

# Un esempio di uso del box model (1)

Riprendiamo l'esempio:

```
<body>
  <h1>Ricetta: la <span class="piatto">pasta al pomodoro</span></h1>
  <p>Vediamo ora come preparare un'ottima <span class="piatto">pasta
  al pomodoro.</span> Innanzitutto, mettere a bollire in una
  pentola capiente abbondante acqua.</p>
  <p>Tagliare a fettine sottili la <span class="ingr">cipolla</span>
  e soffriggerla con l'<span class="ingr">olio</span> in una
  padella fino a quando non sar&agrave; ben dorata. In seguito
  aggiungere i <span class="ingr">pomodori</span> e il
  <span class="ingr">basilico</span>.</p>
  <p>Quando bolle l'acqua, salarla e versare gli <span class="ingr">
  spaghetti</span>. Una volta che gli spaghetti sono cotti,
  scolarli e aggiungervi il sugo di pomodoro.</p>
</body>
```

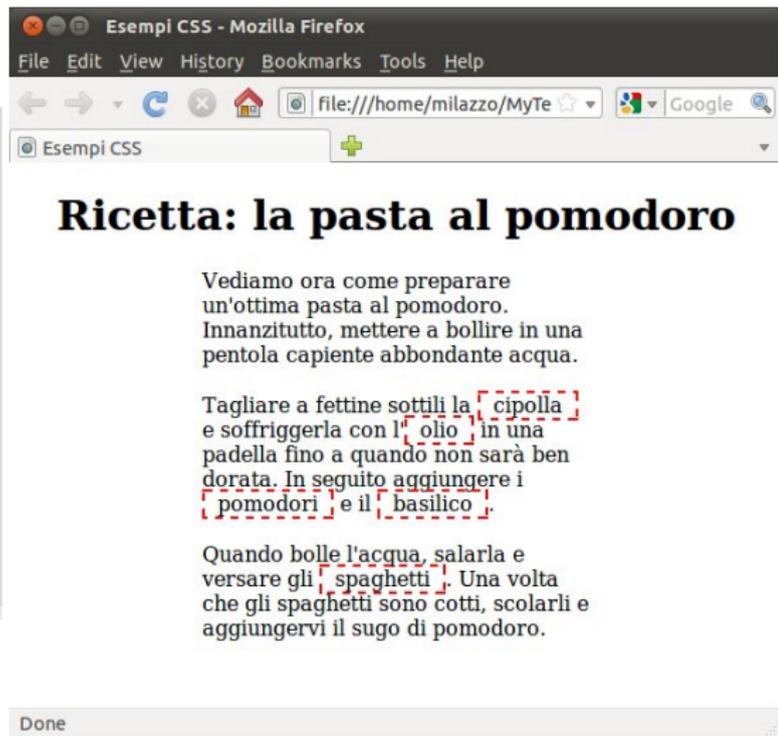
# Un esempio di uso del box model (2)

Il foglio di stile:

```
h1
{
  text-align: center;
}

p
{
  width: 300px;
  margin: 20px auto;
}

span.ingr
{
  border: 2px dashed red;
  padding: 0px 10px;
}
```



The screenshot shows a Mozilla Firefox browser window titled "Esempi CSS - Mozilla Firefox". The address bar shows the file path "file:///home/milazzo/MyTe". The page content is a recipe titled "Ricetta: la pasta al pomodoro". The text is styled according to the CSS code shown in the previous block. The title is centered. The paragraphs are centered and have a width of 300px. The word "ingrediente" is highlighted with a red dashed border and padding.

Esempi CSS

## Ricetta: la pasta al pomodoro

Vediamo ora come preparare un'ottima pasta al pomodoro. Innanzitutto, mettere a bollire in una pentola capiente abbondante acqua.

Tagliare a fettine sottili la cipolla e soffriggerla con l'olio in una padella fino a quando non sarà ben dorata. In seguito aggiungere i pomodori e il basilico.

Quando bolle l'acqua, salarla e versare gli spaghetti. Una volta che gli spaghetti sono cotti, scolarli e aggiungere il sugo di pomodoro.

Done

# Il posizionamento degli elementi (1)

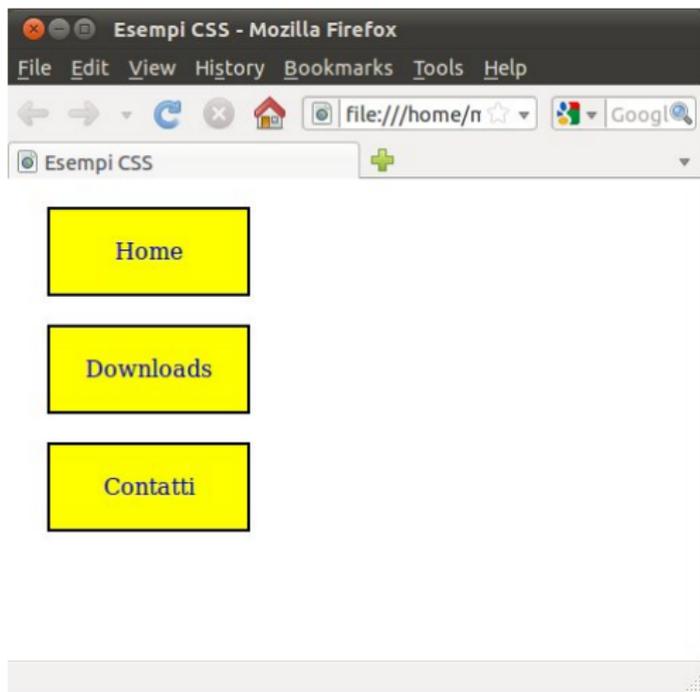
- Il box model consente di descrivere la dimensione, posizione e spaziatura di ogni elemento di un documento HTML
- Il browser legge in sequenza i vari tag del documento HTML e visualizza i contenuti disponendo le corrispondenti “scatole” (box) in uno dei seguenti modi (a seconda del tag usato):
  - ▶ Gli elementi di tipo **block** (es. `div`, `p`, `li`, ecc...) vengono disposti l'uno sotto l'altro in successione verticale
  - ▶ Gli elementi di tipo **inline** (es. `span`, `strong`, ecc...) vengono disposti l'uno accanto all'altro in successione orizzontale
- Inoltre, usando opportune proprietà CSS si può
  - ▶ Visualizzare un elemento di tipo `block` come se fosse `inline` e viceversa
  - ▶ Impostare una visualizzazione **float**, in cui la scatola viene spostata all'estrema sinistra o destra, lasciando che le altre scatole vi girino intorno
  - ▶ Impostare una **posizione assoluta** per una scatola nella finestra del browser (eventualmente nascondendo ciò che già occupava la posizione specificata)

## Il posizionamento degli elementi (2)

- La proprietà `display` consente di impostare la modalità di visualizzazione di un elemento
  - ▶ `block`, `inline`, `none` (elemento non visualizzato), e altri più specifici
- Esempio: menù disposto in orizzontale

```
a
{
  display: block;
  border: 2px solid black;
  background-color: yellow;
  width: 100px;
  margin: 20px;
  padding: 20px;
  text-align: center;
  text-decoration: none;
}
```

```
<a href="#">Home</a>
<a href="#">Downloads</a>
<a href="#">Contatti</a>
```

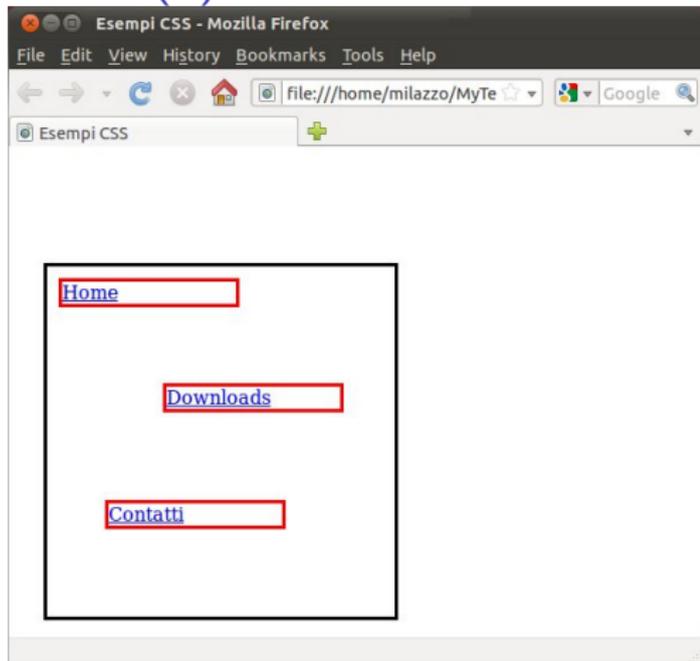


## Il posizionamento degli elementi (3)

- La proprietà `position` consente di specificare il tipo di posizionamento dell'elemento all'interno della finestra del browser.
- Possibili valori:
  - ▶ `static` (valore di default) segue il flusso normale di visualizzazione degli elementi
  - ▶ `relative` consente di specificare la posizione in maniera relativa rispetto al punto in cui l'elemento sarebbe visualizzato di default
  - ▶ `fixed` consente di specificare una posizione fissata relativa alla finestra del browser (l'elemento non viene "scollato via" (rimane fermo) quando si agisce sulla barra di scorrimento)
  - ▶ `absolute` consente di specificare una posizione con coordinate assolute all'interno dello spazio messo a disposizione dall'elemento che lo contiene
- Le posizioni (assolute e relative) sono espresse tramite le proprietà `left`, `top`, `right` e `bottom`

## Il posizionamento degli elementi (4)

```
div#menu
{
  width: 300px;
  height: 300px;
  border: solid;
  position: fixed;
  top: 100px;
  left: 30px;
}
div#menu a
{
  border: solid red;
  width: 150px;
  position: absolute;
}
a#a1 {
  top: 10px;
  left: 10px;
}
a#a2 {
  top: 100px;
  left: 100px;
}
a#a3 {
  top: 200px;
  left: 50px;
}
```



```
<div id="menu">
  <a id="a1" href="...">Home</a>
  <a id="a2" href="...">Downloads</a>
  <a id="a3" href="...">Contatti</a>
</div>
```

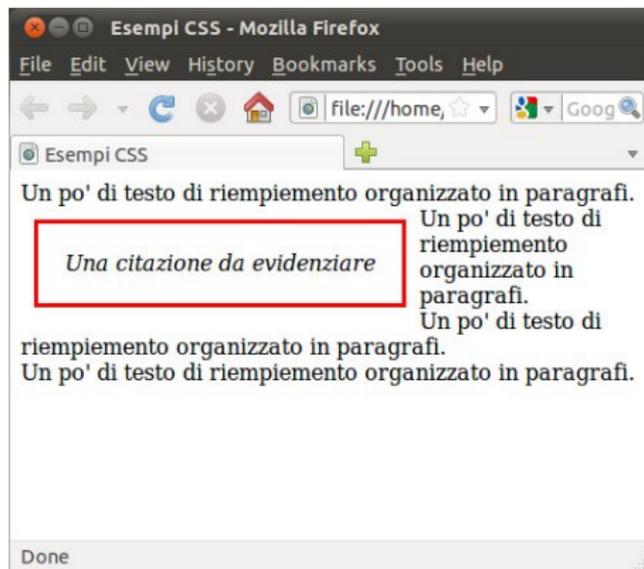
## Il posizionamento degli elementi (5)

- Con CSS è possibile anche “accumulare” elementi sulla sinistra (o sulla destra) usando la proprietà `float`
- Impostando `float:left`; l’elemento viene visualizzato all’estrema sinistra, con gli altri elementi disposti attorno ad esso
- Se si setta la proprietà `float` per più elementi questi vengono “impilati”, sulla sinistra (o sulla destra), eventualmente disposti su più righe
- Gli elementi si ridispongono, se necessario, quando si ridimensiona la finestra del browser
- Per evitare che gli altri elementi del documento HTML si dispongano attorno all’elemento floating si può usare la proprietà `clear`

# Il posizionamento degli elementi (6)

Un esempio di floating:

```
p
{
  margin: 0px;
}
p.citazione
{
  float: left;
  border: solid red;
  font-style: italic;
  margin: 10px;
  padding: 20px;
}
```



applicato a:

```
<p>Un po' di testo di riempimento organizzato in paragrafi</p>
<p class="citazione">Una citazione da evidenziare </p>
<p>Un po' di testo di riempimento organizzato in paragrafi</p>
<p>Un po' di testo di riempimento organizzato in paragrafi</p>
<p>Un po' di testo di riempimento organizzato in paragrafi</p>
```

## Il posizionamento degli elementi (7)

Altri esempi di uso di floating:

- Il sito web di questo corso include un menù disposto alla sinistra del documento usando la proprietà `float`
- Il corso CSS su <http://www.w3schools.com> nella sezione “CSS floating” mostra alcuni esempi interessanti sull'uso di `float`