

TECNOLOGIE WEB

Dove valgono le dichiarazioni CSS ?

DOVE VALE UNA IMPOSTAZIONE CSS?

In alcuni casi è facile. 

```
25. <hr style="height:2px; background-color: blue">
26.
27. <p style="font-size:20px; font-family:Monospace">
28.   Circa il 70% del nostro pianeta è ricoperto d'acqua.
29.   <br/>
30.   L'acqua è l'unica sostanza che si trova in natura,
31.   <i>a temperatura ambiente</i>,
32.   contemporaneamente nei <b>tre</b> stati di aggregazione:
33. </p>
```

Prendiamo un elemento vuoto come questo **hr** 

È chiaro che queste due dichiarazioni valgono solo per quella linea orizzontale. 


DOVE VALE UNA IMPOSTAZIONE CSS?

Quando l'elemento non è vuoto, le impostazioni valgono per tutto il suo contenuto. 

```


25. <hr style="height:2px; background-color: blue"/>
26.
27. <p style="font-size:20px; font-family:Monospace">
28.   Circa il 70% del nostro pianeta è ricoperto d'acqua.
29.   <br/>
30.   L'acqua è l'unica sostanza che si trova in natura,
31.   <i>a temperatura ambiente</i>,
32.   contemporaneamente nei <b>tre</b> stati di aggregazione:
33. </p>

```

Tutto il contenuto di questo paragrafo
è stampato con un carattere **monospace** e con una **dimensione** di 20 pixel. 

Anche gli elementi **i** e **b** (annidati in **p**) 


DOVE VALE UNA IMPOSTAZIONE CSS?


Quindi le impostazioni dichiarate per il **body**
valgono per tutti gli elementi in esso annidati 

```

07. <body style = " font-family:sans-serif;
08.   color:red;
09.   width:500px;
10.   background-color: yellow;
11.   margin:auto;">

```

Ma non tutte le proprietà sono applicabili a tutti gli elementi. 

Così, per esempio in questo caso ,
stiamo dicendo che tutti i testi (**p**, **ol**, **ul**, **h1**, **h2**, ..)
dovranno essere **sans serif** e in **rosso**. 

DOVE VALE UNA IMPOSTAZIONE CSS?

Osserviamo questa pagina web.



Il "Software libero" è software che rispetta la libertà degli utenti e la comunità. In breve, significa che **gli utenti hanno la libertà di eseguire, copiare, distribuire, studiare, modificare e migliorare il software**. Quindi è una questione di libertà, non di prezzo. Per capire il concetto, bisognerebbe pensare alla "libertà di parola" e non alla "birra gratis"; in inglese a volte usiamo "libre" per disambiguare.

Noi difendiamo attivamente queste libertà, perché tutti hanno diritto ad averle. Tramite queste libertà gli utenti (individualmente o nel loro complesso) controllano il programma e le sue funzioni. Quando non sono gli utenti a controllare il programma, allora il programma (che in quel caso chiamiamo "non libero" o "proprietario") controlla gli utenti; e gli sviluppatori controllano il programma, che quindi diventa uno strumento di abuso.

Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:

- Libertà di eseguire il programma come si desidera, per qualsiasi scopo (*libertà 0*).
- Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (*libertà 1*). L'accesso al codice sorgente ne è un prerequisito.
- Libertà di ridistribuire copie in modo da aiutare il prossimo (*libertà 2*).
- Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (*libertà 3*). L'accesso al codice sorgente ne è un prerequisito.

Un programma è software libero se l'utente ha tutte queste libertà in modo adeguato. Altrimenti diciamo che è non libero.

Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il `copyleft`, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.

due immagini
della stessa dimensione

cinque paragrafi
con le stesse impostazioni

quattro punti di un elenco puntato
simili ai paragrafi ma di colore
diverso

L'unico simpatico è il titolo

Come possiamo
sistemare le
impostazioni CSS?



DIAPOSITIVA 5

ALESSANDRO URSOMANDO

L'ELEMENTO STYLE

Cominciamo dalle immagini che hanno un'altezza di 70 pixel.



Il "Software libero" è software che rispetta la libertà degli utenti e la comunità. In breve, significa che **gli utenti hanno la libertà di eseguire, copiare, distribuire, studiare, modificare e migliorare il software**. Quindi è una questione di libertà, non di prezzo. Per capire il concetto, bisognerebbe pensare alla "libertà di parola" e non alla "birra gratis"; in inglese a volte usiamo "libre" per disambiguare.

Noi difendiamo attivamente queste libertà, perché tutti hanno diritto ad averle. Tramite queste libertà gli utenti (individualmente o nel loro complesso) controllano il programma e le sue funzioni. Quando non sono gli utenti a controllare il programma, allora il programma (che in quel caso chiamiamo "non libero" o "proprietario") controlla gli utenti; e gli sviluppatori controllano il programma, che quindi diventa uno strumento di abuso.

Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:

- Libertà di eseguire il programma come si desidera, per qualsiasi scopo (*libertà 0*).
- Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (*libertà 1*). L'accesso al codice sorgente ne è un prerequisito.
- Libertà di ridistribuire copie in modo da aiutare il prossimo (*libertà 2*).
- Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (*libertà 3*). L'accesso al codice sorgente ne è un prerequisito.

Un programma è software libero se l'utente ha tutte queste libertà in modo adeguato. Altrimenti diciamo che è non libero.

Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il `copyleft`, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.

Se non voglio
ripetere due volte
la stessa impostazione posso
usare l'elemento **style**.



```
01. <head>
02.   <title>Software Libero</title>
03.   <meta charset="UTF-8" />
04.   <style>
05.     img {
06.       height:70px;
07.     }
08.   </style>
09. </head>
```

L'elemento **style**
va annidato dentro **head**
e contiene delle **regole CSS**.



DIAPOSITIVA 6

ALESSANDRO URSOMANDO

UNA REGOLA CSS

Cominciamo dalle immagini che hanno un'altezza di 70 pixel.



Il "Software libero" è software che rispetta la libertà degli utenti e la comunità. In breve, significa che **gli utenti hanno la libertà di eseguire, copiare, distribuire, studiare, modificare e migliorare il software**. Quindi è una questione di libertà, non di prezzo. Per capire il concetto, bisognerebbe pensare alla "libertà di parola" e non alla "birra gratis", in inglese a volte usiamo "libre" per disambiguare.

Noi difendiamo attivamente queste libertà, perché tutti hanno diritto ad averle. Tramite queste libertà gli utenti (individualmente o nel loro complesso) controllano il programma e le sue funzioni. Quando non sono gli utenti a controllare il programma, allora il programma (che in quel caso chiamiamo "non libero" o "proprietario") controlla gli utenti; e gli sviluppatori controllano il programma, che quindi diventa uno strumento di abuso.

Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:

- Libertà di eseguire il programma come si desidera, per qualsiasi scopo (*libertà 0*).
- Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (*libertà 1*). L'accesso al codice sorgente ne è un prerequisito.
- Libertà di ridistribuire copie in modo da aiutare il prossimo (*libertà 2*).
- Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (*libertà 3*). L'accesso al codice sorgente ne è un prerequisito.

Un programma è software libero se l'utente ha tutte queste libertà in modo adeguato. Altrimenti diciamo che è non libero.

Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il **copyleft**, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.

Se non voglio ripetere due volte la stessa impostazione posso usare l'elemento **style**.

```
01. <head>
02.   <title>Software Libero</title>
03.   <meta charset="UTF-8" />
04.   <style>
05.     img {
06.       height:70px;
07.     }
08.   </style>
09. </head>
```

L'elemento **style** va annidato dentro **head** e contiene delle **regole CSS**.

UNA REGOLA CSS

Cominciamo dalle immagini che hanno un'altezza di 70 pixel.



Una regola CSS è formata da
il **selettore**
le **parentesi graffe**
una o più **dichiarazioni**

Il "Software libero" è software che rispetta la libertà degli utenti e la comunità. In breve, significa che **gli utenti hanno la libertà di eseguire, copiare, distribuire, studiare, modificare e migliorare il software**. Quindi è una questione di libertà, non di prezzo. Per capire il concetto, bisognerebbe pensare alla "libertà di parola" e non alla "birra gratis", in inglese a volte usiamo "libre" per disambiguare.

Noi difendiamo attivamente queste libertà, perché tutti hanno diritto ad averle. Tramite queste libertà gli utenti (individualmente o nel loro complesso) controllano il programma e le sue funzioni. Quando non sono gli utenti a controllare il programma, allora il programma (che in quel caso chiamiamo "non libero" o "proprietario") controlla gli utenti; e gli sviluppatori controllano il programma, che quindi diventa uno strumento di abuso.

Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:

- Libertà di eseguire il programma come si desidera, per qualsiasi scopo (*libertà 0*).
- Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (*libertà 1*). L'accesso al codice sorgente ne è un prerequisito.
- Libertà di ridistribuire copie in modo da aiutare il prossimo (*libertà 2*).
- Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (*libertà 3*). L'accesso al codice sorgente ne è un prerequisito.

Un programma è software libero se l'utente ha tutte queste libertà in modo adeguato. Altrimenti diciamo che è non libero.

Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il **copyleft**, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.


Se non voglio ripetere due volte la stessa impostazione posso usare l'elemento **style**.

```
01. <head>
02.   <title>Software Libero</title>
03.   <meta charset="UTF-8" />
04.   <style>
05.     img
06.     { height:70px;
07.     }
08.   </style>
09. </head>
```

L'elemento **style** va annidato dentro **head** e contiene delle **regole CSS**.

UNA REGOLA CSS

Cominciamo dalle immagini che hanno un'altezza di 70 pixel.



Una regola CSS è formata da
il **selettore**
le **parentesi graffe**
una o più **dichiarazioni**

Il "Software Libero" è un software libero, non di prezzo. Per "libre" per...
Noi difendiamo...
Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:
• Libertà di eseguire il programma come si desidera, per qualsiasi scopo (libertà 0).
• Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (libertà 1). L'accesso al codice sorgente ne è un prerequisito.
• Libertà di ridistribuire copie in modo da aiutare il prossimo (libertà 2).
• Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (libertà 3). L'accesso al codice sorgente ne è un prerequisito.
Un programma è software libero se l'utente ha tutte queste libertà in modo adeguato. Altrimenti diciamo che è non libero.
Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il **copyleft**, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.


Se non voglio ripetere due volte la stessa impostazione posso usare l'elemento **style**.

```
01. <head>
02.   <title>Software Libero</title>
03.   <meta charset="UTF-8" />
04.   <style>
05.     img {
06.       height:70px;
07.     }
08.   </style>
09. </head>
```

L'elemento **style** va annidato dentro **head** e contiene delle **regole CSS**.

UNA REGOLA CSS

Cominciamo dalle immagini che hanno un'altezza di 70 pixel.



Una regola CSS è formata da
il **selettore**
le **parentesi graffe**
una o più **dichiarazioni**

Il "Software Libero" è un software libero, non di prezzo. Per "libre" per...
Noi difendiamo...
Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:
• Libertà di eseguire il programma come si desidera, per qualsiasi scopo (libertà 0).
• Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (libertà 1). L'accesso al codice sorgente ne è un prerequisito.
• Libertà di ridistribuire copie in modo da aiutare il prossimo (libertà 2).
• Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (libertà 3). L'accesso al codice sorgente ne è un prerequisito.
Un programma è software libero se l'utente ha tutte queste libertà in modo adeguato. Altrimenti diciamo che è non libero.
Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il **copyleft**, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.

Se non voglio ripetere due volte la stessa impostazione posso usare l'elemento **style**.

```
01. <head>
02.   <title>Software Libero</title>
03.   <meta charset="UTF-8" />
04.   <style>
05.     img {
06.       height:70px;
07.     }
08.   </style>
09. </head>
```

L'elemento **style** va annidato dentro **head** e contiene delle **regole CSS**.

L'ELEMENTO STYLE

Torniamo alla nostra pagina WEB e consideriamo il titolo.



Il "Software libero" è software che rispetta la libertà degli utenti e la comunità. In breve, significa che **gli utenti hanno la libertà di eseguire, copiare, distribuire, studiare, modificare e migliorare il software**. Quindi è una questione di libertà, non di prezzo. Per capire il concetto, bisognerebbe pensare alla "libertà di parola" e non alla "birra gratis"; in inglese a volte usiamo "libre" per disambiguare.

Noi difendiamo attivamente queste libertà, perché tutti hanno diritto ad averle. Tramite queste libertà gli utenti (individualmente o nel loro complesso) controllano il programma e le sue funzioni. Quando non sono gli utenti a controllare il programma, allora il programma (che in quel caso chiamiamo "non libero" o "proprietario") controlla gli utenti; e gli sviluppatori controllano il programma, che quindi diventa uno strumento di abuso.

Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:

- Libertà di eseguire il programma come si desidera, per qualsiasi scopo (*libertà 0*).
- Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (*libertà 1*). L'accesso al codice sorgente ne è un prerequisito.
- Libertà di ridistribuire copie in modo da aiutare il prossimo (*libertà 2*).
- Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (*libertà 3*). L'accesso al codice sorgente ne è un prerequisito.

Un programma è software libero se l'utente ha tutte queste libertà in modo adeguato. Altrimenti diciamo che è non libero.

Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il `copyleft`, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.

Poiché c'è un solo elemento **h1** potremmo anche impostare le sue caratteristiche direttamente nell'elemento con l'**attributo style**.

Tuttavia preferiamo farlo nell'**elemento style**.

```
05. img {
06.     height:70px;
07. }
08. h1{
09.     color: steelblue;
10.     text-shadow: lightgrey 2px 2px ;
11.     font-family: fantasy;
12. }
```

L'ELEMENTO STYLE

Consideriamo adesso i paragrafi e l'elenco puntato



Il "Software libero" è software che rispetta la libertà degli utenti e la comunità. In breve, significa che **gli utenti hanno la libertà di eseguire, copiare, distribuire, studiare, modificare e migliorare il software**. Quindi è una questione di libertà, non di prezzo. Per capire il concetto, bisognerebbe pensare alla "libertà di parola" e non alla "birra gratis"; in inglese a volte usiamo "libre" per disambiguare.

Noi difendiamo attivamente queste libertà, perché tutti hanno diritto ad averle. Tramite queste libertà gli utenti (individualmente o nel loro complesso) controllano il programma e le sue funzioni. Quando non sono gli utenti a controllare il programma, allora il programma (che in quel caso chiamiamo "non libero" o "proprietario") controlla gli utenti; e gli sviluppatori controllano il programma, che quindi diventa uno strumento di abuso.

Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:

- Libertà di eseguire il programma come si desidera, per qualsiasi scopo (*libertà 0*).
- Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (*libertà 1*). L'accesso al codice sorgente ne è un prerequisito.
- Libertà di ridistribuire copie in modo da aiutare il prossimo (*libertà 2*).
- Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (*libertà 3*). L'accesso al codice sorgente ne è un prerequisito.

Un programma è software libero se l'utente ha tutte queste libertà in modo adeguato. Altrimenti diciamo che è non libero.

Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il `copyleft`, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.

Invece di ripetere cento volte le stesse impostazioni inseriamo nell'**elemento style** queste tre regole.

```
01. p,li {
02.     font-family:sans-serif;
03.     text-align: justify;
04. }
05. p{
06.     color: midnightblue;
07. }
08. li{
09.     color: steelblue;
10. }
```

Anche **questo** è un selettore valido

LE DIMENSIONI DEI TESTI

Non resta che impostare la dimensione dei testi (**h1**, **p** e **li**)

Software Libero

Il "Software libero" è software che rispetta la libertà degli utenti e la comunità. In breve, significa che **gli utenti hanno la libertà di eseguire, copiare, distribuire, studiare, modificare e migliorare il software**. Quindi è una questione di libertà, non di prezzo. Per capire il concetto, bisognerebbe pensare alla "libertà di parola" e non alla "birra gratis"; in inglese a volte usiamo "libre" per disambiguare.

Noi difendiamo attivamente queste libertà, perché tutti hanno diritto ad averle. Tramite queste libertà gli utenti (individualmente o nel loro complesso) controllano il programma e le sue funzioni. Quando non sono gli utenti a controllare il programma, allora il programma (che in quel caso chiamiamo "non libero" o "proprietario") controlla gli utenti; e gli sviluppatori controllano il programma, che quindi diventa uno strumento di abuso.

Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:

- Libertà di eseguire il programma come si desidera, per qualsiasi scopo (*libertà 0*).
- Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (*libertà 1*). L'accesso al codice sorgente ne è un prerequisito.
- Libertà di ridistribuire copie in modo da aiutare il prossimo (*libertà 2*).
- Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (*libertà 3*). L'accesso al codice sorgente ne è un prerequisito.

Un programma è software libero se l'utente ha tutte queste libertà in modo adeguato. Altrimenti diciamo che è non libero.

Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il `copyleft`, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.

La proprietà da utilizzare in questo caso, già lo sappiamo, è **font-size**.

Aggiungiamo alle due regole che abbiamo visto prima le dichiarazioni CSS opportune.

```

10. h1{
11.     color: steelblue;
12.     text-shadow: lightgrey 2px 2px 0;
13.     font-family: fantasy;
14.     font-size: 75px;
15. }
16. p,li {
17.     font-family:sans-serif;
18.     text-align: justify;
19.     font-size: 15px;
20. }
```

UNA STRATEGIA DIFFUSA PER LA DIMENSIONE DEI TESTI

In realtà la strategia che viene adoperata è la seguente.

Software Libero

Il "Software libero" è software che rispetta la libertà degli utenti e la comunità. In breve, significa che **gli utenti hanno la libertà di eseguire, copiare, distribuire, studiare, modificare e migliorare il software**. Quindi è una questione di libertà, non di prezzo. Per capire il concetto, bisognerebbe pensare alla "libertà di parola" e non alla "birra gratis"; in inglese a volte usiamo "libre" per disambiguare.

Noi difendiamo attivamente queste libertà, perché tutti hanno diritto ad averle. Tramite queste libertà gli utenti (individualmente o nel loro complesso) controllano il programma e le sue funzioni. Quando non sono gli utenti a controllare il programma, allora il programma (che in quel caso chiamiamo "non libero" o "proprietario") controlla gli utenti; e gli sviluppatori controllano il programma, che quindi diventa uno strumento di abuso.

Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:

- Libertà di eseguire il programma come si desidera, per qualsiasi scopo (*libertà 0*).
- Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (*libertà 1*). L'accesso al codice sorgente ne è un prerequisito.
- Libertà di ridistribuire copie in modo da aiutare il prossimo (*libertà 2*).
- Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (*libertà 3*). L'accesso al codice sorgente ne è un prerequisito.

Un programma è software libero se l'utente ha tutte queste libertà in modo adeguato. Altrimenti diciamo che è non libero.

Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il `copyleft`, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.

Innanzitutto si imposta una dimensione assoluta (in **pixel**) per il **body**.

```

10. body {
11.     font-size: 15px;
12. }
```

Successivamente, si impostano le dimensioni di tutti i testi in una unità di misura **relativa** alla misura stabilita per il body: **em**. In questo caso:

1em = 15px	5em = 75px
2em = 30px	2.5em = 37.5px

UNA STRATEGIA DIFFUSA PER LA DIMENSIONE DEI TESTI

In realtà la strategia che viene adoperata è la seguente.



Il "Software libero" è software che rispetta la libertà degli utenti e la comunità. In breve, significa che **gli utenti hanno la libertà di eseguire, copiare, distribuire, studiare, modificare e migliorare il software**. Quindi è una questione di libertà, non di prezzo. Per capire il concetto, bisognerebbe pensare alla "libertà di parola" e non alla "birra gratis"; in inglese a volte usiamo "libre" per disambiguare.

Noi difendiamo attivamente queste libertà, perché tutti hanno diritto ad averle. Tramite queste libertà gli utenti (individualmente o nel loro complesso) controllano il programma e le sue funzioni. Quando non sono gli utenti a controllare il programma, allora il programma (che in quel caso chiamiamo "non libero" o "proprietario") controlla gli utenti; e gli sviluppatori controllano il programma, che quindi diventa uno strumento di abuso.

Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:

- Libertà di eseguire il programma come si desidera, per qualsiasi scopo (*libertà 0*).
- Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (*libertà 1*). L'accesso al codice sorgente ne è un prerequisito.
- Libertà di ridistribuire copie in modo da aiutare il prossimo (*libertà 2*).
- Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (*libertà 3*). L'accesso al codice sorgente ne è un prerequisito.

Un programma è software libero se l'utente ha tutte queste libertà in modo adeguato. Altrimenti diciamo che è non libero.

Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il `copyleft`, noto anche impropriamente come "permesso d'autore" (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.

```
10. body {
11.     font-size: 15px;
12. }
```

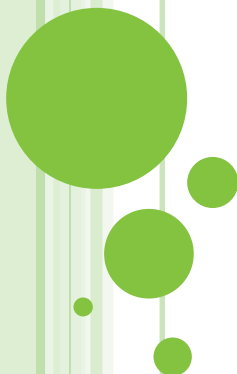
```
13. h1{
14.     color: steelblue;
15.     text-shadow: lightgrey 2px 2px 0;
16.     font-family: fantasy;
17.     font-size: 5em;
18. }
19. p,li {
20.     font-family:sans-serif;
21.     text-align: justify;
22.     font-size: 1em;
23. }
```

Questa strategia
– che gestisce il rapporto tra le
dimensioni dei testi –
ci permette di rivoluzionare
agilmente l'aspetto del sito.



TECNOLOGIE WEB

Path assoluti e relativi, remoti e locali



DEFINIZIONI

Un **path** (o anche **percorso**)
è l'elenco delle cartelle da visitare per raggiungere una certa risorsa.

Un **path assoluto**
indica il percorso dall'inizio.

Un **path relativo**
indica il percorso da una certa cartella.

path	assoluto	relativo
<code>C:/Cicero/Cicero_SVN/htdocs/css/images/overlay.png</code>	<input checked="" type="checkbox"/>	
<code>../../immagini/sfondo.png</code>		<input checked="" type="checkbox"/>
<code>http://www.bbuio.it/didattica/index.html</code>	<input checked="" type="checkbox"/>	
<code>immagini/miniature/prodotto.png</code>		<input checked="" type="checkbox"/>

DEFINIZIONI

Un **path** (o anche **percorso**)
è l'elenco delle cartelle da visitare per raggiungere una certa risorsa.

Un **path assoluto**
indica il percorso dall'inizio.

Un **path relativo**
indica il percorso da una certa cartella.

Un **path** (assoluto o relativo che sia) può essere **locale** o **remoto**.

Un **path locale**
è relativo al nostro computer.

Un **path remoto**
è relativo ad una risorsa online.

path	assoluto	relativo	locale	remoto
<code>C:/Cicero/Cicero_SVN/htdocs/css/images/overlay.png</code>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	
<code>../../immagini/sfondo.png</code>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<code>http://www.bbuio.it/didattica/index.html</code>	<input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>
<code>immagini/miniature/prodotto.png</code>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

RIASSUMENDO

Se il path è **assoluto** è facile capire se è remoto o locale. 

Se è **remoto** comincia con l'indicazione del protocollo 


Se è **locale** comincia dall'inizio del file system: 

- / per Linux e Mac OS
- il nome dell'unità sotto Windows (c:/)

path	assoluto	relativo	locale	remoto
<code>C:/Cicero/Cicero_SVN/htdocs/css/images/overlay.png</code>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	
<code>.././immagini/sfondo.png</code>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<code>http://www.bbuio.it/didattica/index.html</code>	<input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>
<code>immagini/miniature/prodotto.png</code>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

RIASSUMENDO

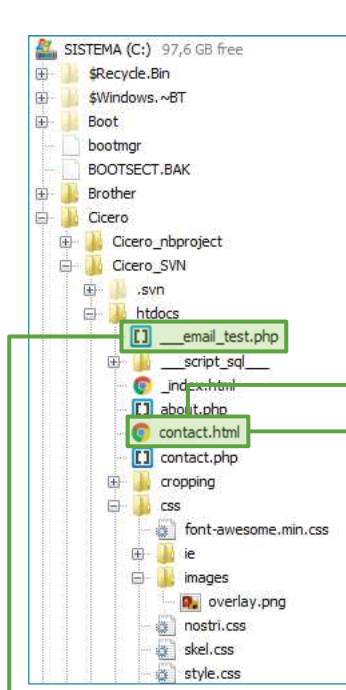
Se è **relativo** invece: non sappiamo se è locale o remoto 

O meglio non ci interessa! 

Vediamo qualche esempio. 

path	assoluto	relativo	locale	remoto
<code>C:/Cicero/Cicero_SVN/htdocs/css/images/overlay.png</code>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	
<code>.././immagini/sfondo.png</code>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<code>http://www.bbuio.it/didattica/index.html</code>	<input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>
<code>immagini/miniature/prodotto.png</code>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

PATH RELATIVI



Consideriamo il **path assoluto** di questo file:

```
C:\Cicero\Cicero_SVN\htdocs\contact.html
```

Il **path relativo** di questo stesso file deve essere riferito ad una posizione specifica.

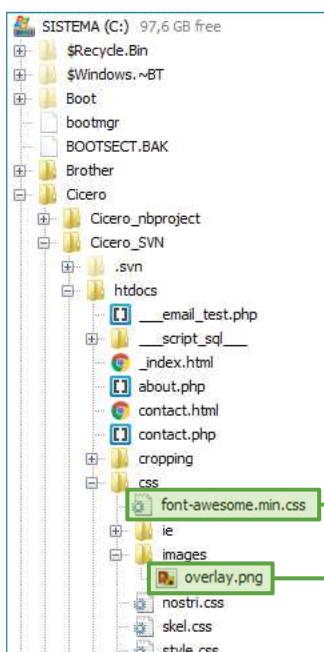
Supponiamo che sia **questo** file a dover fare riferimento ad esso.

Poiché i due file sono nella stessa cartella il **path relativo** coincide con il nome del file.

```
contact.html
```

Non devo attraversare nessuna cartella: non ho nessuna cartella da indicare.

PATH RELATIVI



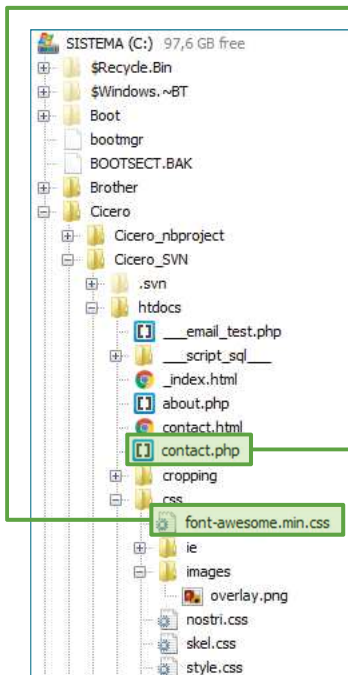
Supponiamo adesso che sia **questo** file..

..a dover fare riferimento a **questo** altro file.

Per poter raggiungere il file devo prima entrare nella cartella **images**, quindi il percorso relativo è:

```
images/overlay.png
```

PATH RELATIVI



Supponiamo infine che sia sempre questo file a dover fare riferimento ad un file che non si trovi però nel suo sottoalbero.

Supponiamo ci si debba riferire a questo file.

Bisognerà prima salire dalla cartella in cui mi trovo (**css**) nella cartella superiore (**htdocs**) e poi fare riferimento al file in questione.

Per risalire l'albero delle cartelle si scrive ..
pertanto il path in questione sarà:

```
../htdocs/contact.php
```

QUALCOSA DI NUOVO

CSS
HTML

QUALCOSA DI NUOVO (CSS)

UNA STRATEGIA DIFFUSA PER LA DIMENSIONE DEI TESTI

In realtà la strategia che viene adoperata è la seguente.

Software Libero

Innanzitutto si imposta una dimensione assoluta (in pixel) per il **body**.

10. body {
11. font-size: 15px;
12. }

13. h1 {
14. color: steelblue;
15. text-shadow: lightgrey 2px 2px 0;
16. font-family: fantasy;
17. font-size: 5em;
18. }
19. p, li {
20. font-family: sans-serif;
21. text-align: justify;
22. font-size: 1em;
23. }

```

13. h1{
14.   color: steelblue;
15.   text-shadow: lightgrey 2px 2px 0;
16.   font-family: fantasy;
17.   font-size: 5em;
18. }
19. p,li {
20.   font-family:sans-serif;
21.   text-align: justify;
22.   font-size: 1em;
23. }
    
```

Dà un'ombra al testo

Specifica il tipo di allineamento

In generale, dove troviamo le **proprietà CSS** e i **valori** che possono assumere?

MANUALETTO CSS

text-align	start end left right center justify
text-align-last	start end left right center justify
text-decoration	none underline overline line-through blink
text-emphasis	none [[accent dot circle disc] [before after]?]
text-indent	length %
text-justify	auto inter-word inter-ideograph inter-cluster distribute kashida tibetan
text-outline	none color length
text-shadow	none color length
text-transform	none capitalize uppercase lowercase
text-wrap	normal unrestricted none suppress

```

13. h1{
14.   color: steelblue;
15.   text-shadow: lightgrey 2px 2px 0;
16.   font-family: fantasy;
17.   font-size: 5em;
18. }
19. p,li {
20.   font-family:sans-serif;
21.   text-align: justify;
22.   font-size: 1em;
23. }
    
```

Dà un'ombra al testo

Specifica il tipo di allineamento

<http://www.bbuio.it/didattica/css/manualetto.pdf>

QUALCOSA DI NUOVO (HTML)



Il "Software libero" è software che rispetta la libertà degli utenti e la comunità. In breve, significa che **gli utenti hanno la libertà di eseguire, copiare, distribuire, studiare, modificare e migliorare il software**. Quindi è una questione di libertà, non di prezzo. Per capire il concetto, bisognerebbe pensare alla "libertà di parola" e non alla "birra gratis"; in inglese a volte usiamo "libre" per disambiguare.

Noi difendiamo attivamente queste libertà, perché tutti hanno diritto ad averle. Tramite queste libertà gli utenti (individualmente o nel loro complesso) controllano il programma e le sue funzioni. Quando non sono gli utenti a controllare il programma, allora il programma (che in quel caso chiamiamo "non libero" o "proprietario") controlla gli utenti, e gli sviluppatori controllano il programma, che quindi diventa uno strumento di abuso.

Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:

- Libertà di eseguire il programma come si desidera, per qualsiasi scopo (*libertà 0*).
- Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (*libertà 1*). L'accesso al codice sorgente ne è un prerequisito.
- Libertà di ridistribuire copie in modo da aiutare il prossimo (*libertà 2*).
- Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (*libertà 3*). L'accesso al codice sorgente ne è un prerequisito.

Un programma è software libero se l'utente ha tutte queste libertà in modo adeguato. Altrimenti diciamo che è non libero.

Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il **copyleft**, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.

QUALCOSA DI NUOVO (HTML)

Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il **copyleft**, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.

Avrai notato **questa** parte della pagina web vista prima.



Avrai intuito che si tratta di un **link**.



L'elemento per ottenere un link è **a**.



L'attributo **href** specifica la risorsa da aprire quando si clicca sul contenuto.

Il **contenuto** può essere un testo, un'immagine o altro.

LINK

Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il `copyleft`, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito, non è possibile aggiungere restrizioni per negare ad altre persone le libertà principali. Questa regola non entra in conflitto con le libertà principali, anzi le protegge.

```

01. <p>
02.   Tuttavia, certi tipi di regole sul come distribuire il software libero
03.   sono accettabili quando non entrano in conflitto con le libertà principali.
04.   Per esempio, il <a href="copyleft.html">copyleft</a>, noto anche
05.   impropriamente come permesso d'autore , e (detto in poche parole)
06.   la regola per cui, quando il programma è ridistribuito, non è possibile
07.   aggiungere restrizioni per negare ad altre persone le libertà principali.
08.   Questa regola non entra in conflitto con le libertà principali,
09.   anzi le protegge.
10. </p>

```

L'elemento per ottenere un link è **a**.

L'attributo **href** specifica la risorsa da aprire quando si clicca sul contenuto.

Il **contenuto** può essere un testo, un'immagine o altro.

Esisterà un simpatico manualetto HTML come quello CSS appena visto?

MANUALETTO HTML

Esiste! Ed è anche migliore!
Per ogni **tag**, ci dice a cosa serve in quali versioni di HTML è valido e l'elenco degli attributi

<code><a></code>	hyperlink	4 / 5	href hreflang media ping rel target type
------------------------	-----------	-------	--

<http://www.bbuio.it/didattica/html/manualetto.pdf>

Esisterà un simpatico manualetto HTML come quello CSS appena visto?

L'ELEMENTO A COME SEGNAPOSTO

Un **segnaposto** è una posizione specifica all'interno di una pagina web indirizzabile da un **link**.



Tutti abbiamo ben presente l'indice di una voce di wikipedia.

L'ELEMENTO A COME SEGNAPOSTO

Un **segnaposto** è una posizione specifica all'interno di una pagina web indirizzabile da un **link**.



Tutti abbiamo ben presente l'indice di una voce di wikipedia.

Tutti sappiamo che cliccando su un punto dell'indice si viene dirottati all'inizio di quella parte di pagina.

Per esempio..

L'ELEMENTO A COME SEGNAPOSTO

Un **segnaposto** è una posizione specifica all'interno di una pagina web indirizzabile da un **link**.



Tutti abbiamo ben presente l'indice di una voce di wikipedia. Tutti sappiamo che cliccando su un punto dell'indice si viene dirottati all'inizio di quella parte di pagina. Per esempio..

L'ELEMENTO A COME SEGNAPOSTO

Un **segnaposto** è una posizione specifica all'interno di una pagina web indirizzabile da un **link**.



Tutti abbiamo ben presente l'indice di una voce di wikipedia. Tutti sappiamo che cliccando su un punto dell'indice si viene dirottati all'inizio di quella parte di pagina. Per esempio.. Per esempio..

L'ELEMENTO A COME SEGNAPOSTO

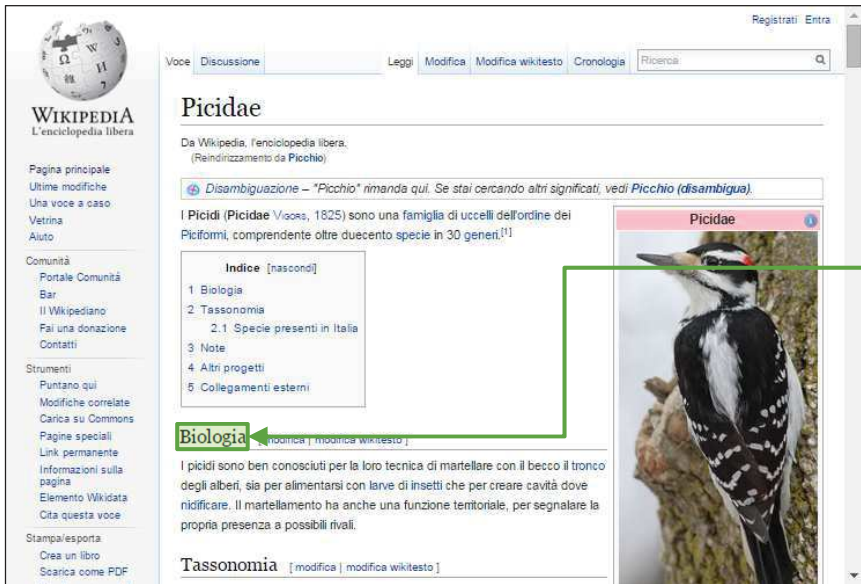
Un **segnaposto** è una posizione specifica all'interno di una pagina web indirizzabile da un **link**.



Tutti abbiamo
l'indice di
wikipedia.
Cliccando su un
indice si viene
all'inizio di
la pagina.
Per esempio..

L'ELEMENTO A COME SEGNAPOSTO

Un **segnaposto** è una posizione specifica all'interno di una pagina web indirizzabile da un **link**.



Per realizzare un segnaposto si deve innanzitutto dare un nome al posto che si vuole indicare.

Per far ciò si usa l'attributo globale **id** (identification) che viene valorizzato con un valore univoco nella pagina.

L'ELEMENTO A COME SEGNAPOSTO

Un **segnaposto** è una posizione specifica all'interno di una pagina web indirizzabile da un **link**.



```
...
<h2 id="voce_biol">Biologia</h2>
...
```

Fatto questo, si valorizza l'attributo **href** dell'elemento **a** in questione con l'**id** dell'elemento obiettivo preceduto dal simbolo **#**

```
...
<ol>
  <li>
    <a href="#voce_biol">
      Biologia
    </a>
  </li>
...

```

