

TECNOLOGIE WEB

Integrare HTML e CSS
Reset

CC BY

INTEGRAZIONE HTML - CSS

Per prima cosa abbiamo visto le **dichiarazioni in linea** con l'**attributo style**.

```
25. <hr style="height:2px; background-color: blue"/>
26.
27. <sp 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>
34.
35. <ol style="font-size:20px; font-family:Monospace">
36. <li>Solido</li>
37. <li>Liquido</li>
38. <li>Gassoso</li>
39. </ol>
```

Successivamente abbiamo visto i
fogli di stile interni con l'**elemento style**

Vediamo infine i **fogli di stile esterni**

```
01. <style>
02.   img {
03.     height:70px;
04.   }
05.   body {
06.     font-size: 15px;
07.   }
08.   h1{
09.     color: steelblue;
10.     text-shadow: lightgrey 2px 2px;
11.     font-family: fantasy;
12.     font-size: 5em;
13.   }
14.   p,li {
15.     font-family:sans-serif;
16.     text-align: justify;
17.     font-size: 1em;
18.   }
19.   p{
20.     color: midnightblue;
21.   }
22.   li{
23.     color: steelblue;
24.   }
25. </style>
```

FOGLI DI STILE ESTERNI

L'elemento style qui sotto è stupidamente identico per queste due pagine 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.



Il copyleft è un metodo per dire a tutti: "Questo è un software libero e guai a chi ci mette le mani sopra!"

Quindi rilasciando un'opera (non necessariamente un software) in **copyleft** conferiamo alla stessa le quattro libertà fondamentali del software libero ma contemporaneamente impediamo a chiunque decida di migliorarlo di diffondere le sue estensioni del nostro lavoro come opere proprietarie.

Questo meccanismo rappresenta inoltre un incentivo per i programmatori a far crescere il software libero. Alcuni

```
01. <style>
02.   img {
03.     height:70px;
04.   }
05.   body {
06.     font-size: 15px;
07.   }
08.   h1{
09.     color: steelblue;
10.     text-shadow: lightgrey 2px 2px;
11.     font-family: fantasy;
12.     font-size: 5em;
13.   }
14.   p,li {
15.     font-family:sans-serif;
16.     text-align: justify;
17.     font-size: 1em;
18.   }
19.   p{
20.     color: midnightblue;
21.   }
22.   li{
23.     color: steelblue;
24.   }
25. </style>
```

FOGLI DI STILE ESTERNI

Quando progettiamo un sito vogliamo che tutte le pagine abbiano lo stesso aspetto.



stili.css

```
01.   img {
02.     height:70px;
03.   }
04.   body {
05.     font-size: 15px;
06.   }
07.   h1{
08.     color: steelblue;
09.     text-shadow: lightgrey 2px 2px;
10.     font-family: fantasy;
11.     font-size: 5em;
12.   }
13.   p,li {
14.     font-family:sans-serif;
15.     text-align: justify;
16.     font-size: 1em;
17.   }
18.   p{
19.     color: midnightblue;
20.   }
21.   li{
22.     color: steelblue;
23.   }
```

Per ottenere questo

realizziamo un file con tutte le regole CSS e linkiamo quel file in tutte le pagine html del sito.



index.html

```
01. <link type="text/css" rel="stylesheet" href="stili.css" />
```

copyleft.html

```
01. <link type="text/css" rel="stylesheet" href="stili.css" />
```

L'elemento (vuoto) **link** è annidato in **head**.

Noi lo usiamo con gli attributi **type** e **rel** valorizzati come in figura e con l'attributo **href** che indica il path della risorsa da linkare.



TRE POSSIBILITÀ: QUALE SCEGLIERE?

FOGLI DI STILE ESTERNI

FOGLI DI STILE INTERNI

DICHIARAZIONI IN LINEA

Il buon senso
ci spinge ad usare
SEMPRE
i fogli di stile esterni.



TRE POSSIBILITÀ: QUALE SCEGLIERE?

FOGLI DI STILE ESTERNI

FOGLI DI STILE INTERNI

DICHIARAZIONI IN LINEA

Si possono usare
le tre strategie insieme?



Quali sono le priorità ?



TRE POSSIBILITÀ: QUALE SCEGLIERE?

1. **DEFAULT DEL BROWSER**
2. **FOGLI DI STILE ESTERNI**
3. **FOGLI DI STILE INTERNI**
4. **DICHIARAZIONI IN LINEA**

Le pagine vengono renderizzate dal browser seguendo questo ordine di priorità



LE SCELTE DEI BROWSER: CONSEGUENZE

1. **DEFAULT DEL BROWSER**
2. **FOGLI DI STILE ESTERNI**
3. **FOGLI DI STILE INTERNI**
4. **DICHIARAZIONI IN LINEA**

- lo stesso documento HTML viene mostrato in maniera diversa da browser diversi
- il mio documento HTML viene mostrato con delle caratteristiche che non ho scelto io consapevolmente.



LE SCELTE DEI BROWSER: CONSEGUENZE

1. DEFAULT DEL BROWSER

2. FOGLI DI STILE ESTERNI

3. FOGLI DI STILE INTERNI

4. DICHIARAZIONI IN LINEA



Eric Meyer ha pensato di **neutralizzare** tutte le scelte di default fatte dai browser con un **file .css**.

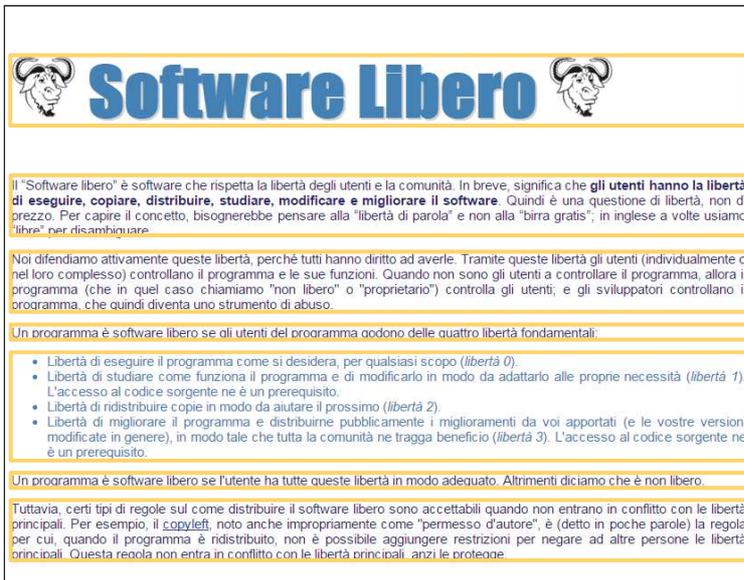
<http://meyerweb.com/eric/tools/css/reset/>

TECNOLOGIE WEB

Gli elementi contenitori
I valori percentuali (delle proprietà)

GLI ELEMENTI CONTENITORI

Cominciamo con l'osservare questa pagina web.



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.

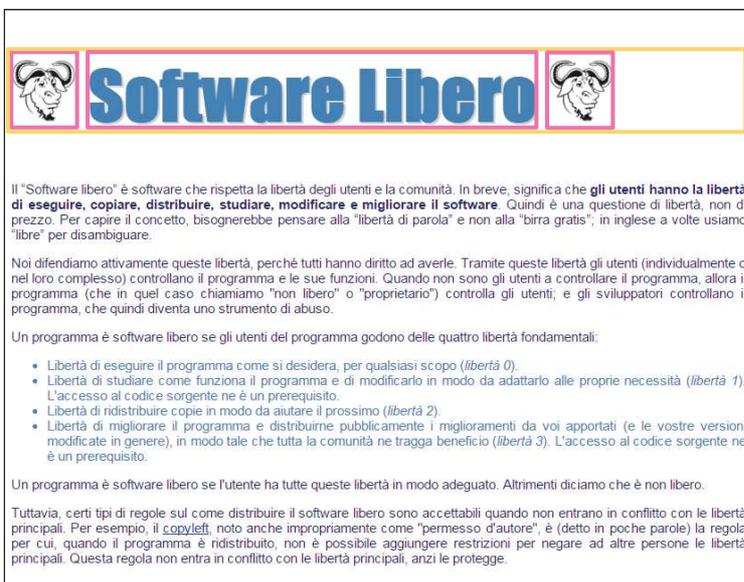
Tuttavia, certi tipi di regole sul come distribuire il software libero sono accettabili quando non entrano in conflitto con le libertà principali. Per esempio, il `copyleft`, noto anche impropriamente come "permesso d'autore", è (detto in poche parole) la regola per cui, quando il programma è ridistribuito,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

In questa pagina
l'elemento **body** contiene
gli elementi:
h1, p, p, p, ol, p, p.

Gli elementi
h1, p, p, p, ol, p, p
sono tutti elementi contenitori.

GLI ELEMENTI CONTENITORI

Cominciamo con l'osservare questa pagina web.



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.

In questa pagina
l'elemento **body** contiene
gli elementi:
h1, p, p, p, ol, p, p.

Gli elementi
h1, p, p, p, ol, p, p
sono tutti elementi contenitori.

L'elemento **h1**
nello specifico contiene
un elemento **img**,
del testo
e un altro elemento **img**.

GLI ELEMENTI CONTENITORI

La prima osservazione che facciamo l'abbiamo già vista:
un contenitore può essere usato per dare una specifica CSS a tutto il suo contenuto.



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 dei

- Libertà di eseguire il programma come si desidera.
- Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità.
- L'accesso al codice sorgente ne è un prerequisito.
- Libertà di ridistribuire copie in modo da aiutare il prossimo.
- 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.

Un programma è software libero se l'utente ha tutte

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 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.



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.

```
07. h1{
08.     color: steelblue;
09.     text-shadow: lightgrey 2px 2px;
10.     font-family: fantasy;
11.     font-size: 5em;
12.     text-align: center;
13. }
```

GLI ELEMENTI CONTENITORI

Sappiamo inoltre che essi occupano tutto lo spazio che viene loro concesso.



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.

```
01. width: 100%; // default
```

In effetti,
possiamo indicare la
dimensione di un contenitore
anche con un valore
percentuale
(sullo spazio a disposizione).

Di solito
si stabilisce la dimensione del
body (un valore diffuso è 800px)
e si usano le percentuali per i
suoi contenuti.

GLI ELEMENTI CONTENITORI

Sappiamo inoltre che essi occupano tutto lo spazio che viene loro concesso.





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, nel loro complesso) controllano il programma (che in quel caso chiamiamo programma, che quindi diventa uno strumento).

Un programma è software libero se gli utenti:

- Libertà di eseguire il programma con
- Libertà di studiare come funziona il
- L'accesso al codice sorgente ne è un
- Libertà di ridistribuire copie in modo
- Libertà di migliorare il programma (modificate in genere), in modo tale da un prerequisito.

Un programma è software libero se l'utente

Tuttavia, certi tipi di regole sul come distribuirlo principali. Per esempio, il [copyleft](#), noto anche come "copyleft", è una regola che, quando il programma è ridistribuito, non questa regola non entra in conflitto con le

```

01. body {
02.   width: 800px;
03.   margin: auto;
04. }
05. p {
06.   width: 70%;
07. }

```



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:

to
nensione del
ffuso è 800px)
centuali per i
enuti.

GLI ELEMENTI CONTENITORI

Sappiamo inoltre che essi occupano tutto lo spazio che viene loro concesso.





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, nel loro complesso) controllano il programma (che in quel caso chiamiamo programma, che quindi diventa uno strumento).

Un programma è software libero se gli utenti:

- Libertà di eseguire il programma con
- Libertà di studiare come funziona il
- L'accesso al codice sorgente ne è un
- Libertà di ridistribuire copie in modo
- Libertà di migliorare il programma (modificate in genere), in modo tale da un prerequisito.

Un programma è software libero se l'utente

Tuttavia, certi tipi di regole sul come distribuirlo principali. Per esempio, il [copyleft](#), noto anche come "copyleft", è una regola che, quando il programma è ridistribuito, non questa regola non entra in conflitto con le

```

01. body {
02.   width: 800px;
03.   margin: auto;
04. }
05. p {
06.   width: 70%;
07. }

```



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:

to
nensione del
ffuso è 800px)
centuali per i
enuti.

GLI ELEMENTI CONTENITORI

Sappiamo inoltre che essi occupano tutto lo spazio che viene loro concesso.





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, nel loro complesso) controllano il programma (che in quel caso chiamiamo programma, che quindi diventa uno strumento).

Un programma è software libero se gli utenti

- 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.

```

01. body {
02.     width: 800px;
03.     margin: auto;
04. }
05. p {
06.     width: 70%;
07. }

```



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:

to
ensione del
ffuso è 800px)
centuali per i
enuti.



DIAPOSITIVA 17

ALESSANDRO URSOMANDO

GLI ELEMENTI CONTENITORI

Esistono poi altri contenitori che si usano per accoppiare dei contenuti, il primo è **div**.





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.

```

01. <!doctype html>
02. <html>
03.   <head>
04.     <title>
05.       Software Libero
06.     </title>
07.     <meta charset="UTF-8" />
08.     <link type="text/css"
09.       rel="stylesheet"
10.       href="css/stili.css"/>
11.   </head>
12.   <body>
13.     <div>
14.       ...
15.     </div>
16.   </body>
17. </html>

```

Facciamo in modo
che **div** contenga tutto ciò che
adesso è contenuto da **body**.



DIAPOSITIVA 18

ALESSANDRO URSOMANDO

GLI ELEMENTI CONTENITORI

Esistono poi altri contenitori che si usano per accoppiare dei contenuti, il primo è **div**.




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.

```
01. body {
02.     background-image:
03.         url(../immagini/pattern.png)
04. }
05.
06. div {
07.     width: 800px;
08.     margin: auto;
09.     background-color: white;
10. }
```

Spostiamo
le specifiche di **body** su **div**
e diamo uno sfondo al **body**.



E lo stesso accade
automaticamente per tutte
le pagine del sito.



GLI ELEMENTI CONTENITORI

Esistono poi altri contenitori che si usano per accoppiare dei contenuti, il primo è **div**.




Il **copyleft** è un metodo per dire a tutti: "Questo è un [software libero](#) e guai a chi ci mette le mani sopra!"

Quindi rilasciando un'opera (non necessariamente un software) in **copyleft** conferiamo alla stessa le quattro libertà fondamentali del [software libero](#), ma contemporaneamente impediamo a chiunque decida di [migliorarlo](#) di diffondere le sue estensioni del nostro lavoro come opere proprietarie.

Questo meccanismo rappresenta inoltre un incentivo per i programmatori a far crescere il [software libero](#). Alcuni programmi liberi importanti, come il compilatore GNU C++, esistono solo grazie a questo.

Alcuni programmatori che lavorano per aziende o università cominciano il proprio lavoro partendo da software rilasciato con questo tipo di licenza proprio per poter contribuire con il loro lavoro al miglioramento del [software libero](#). Ragionevolmente, il datore di lavoro di un programmatore - che sia un ente o un'azienda - vorrebbe lucrare sui prodotti sviluppati dai suoi dipendenti ma alla fine, poiché è illegale distribuire versioni migliorate del [software libero](#) se non come [software libero](#), decidono di rilasciare i prodotti con questa licenza piuttosto che buttarla via tutto.

Rilasciare un'opera d'ingegno in **copyleft** non significa rinunciare alla paternità dell'opera. Il diritto di paternità dell'autore ovviamente rimane, ciò a cui si rinuncia è il diritto al pagamento di un compenso per l'utilizzo di quell'opera. La si regala dunque alla collettività, ma il mittente di questo regalo è evidente e conosciuto. Inoltre la paternità dell'opera resta attaccata all'opera come le quattro libertà fondamentali (che abbiamo visto nella pagina del [software libero](#)).

Spesso si sente parlare di software **open source**. Si tratta di qualcosa di praticamente identico al *free software* (o [software libero](#), in italiano) rilasciato in **copyleft**. La locuzione nacque perché in inglese *free* significa *libero* ma significa anche *gratuito* ed in effetti i servizi connessi ad un software rilasciato a sorgente aperta (**open source**) potrebbero essere a pagamento (i supporti, i manuali, l'assistenza, ecc).

```
01. body {
02.     background-image:
03.         url(../immagini/pattern.png)
04. }
05.
06. div {
07.     width: 800px;
08.     margin: auto;
09.     background-color: white;
10. }
```

Spostiamo
le specifiche di **body** su **div**
e diamo uno sfondo al **body**.



E lo stesso accade
automaticamente per tutte
le pagine del sito.



DIV E I NUOVI CONTENITORI SEMANTICI

Esistono poi altri contenitori che si usano per accorpare dei contenuti, il primo è **div**.



Con HTML5 si sono resi disponibili altri contenitori semantici.



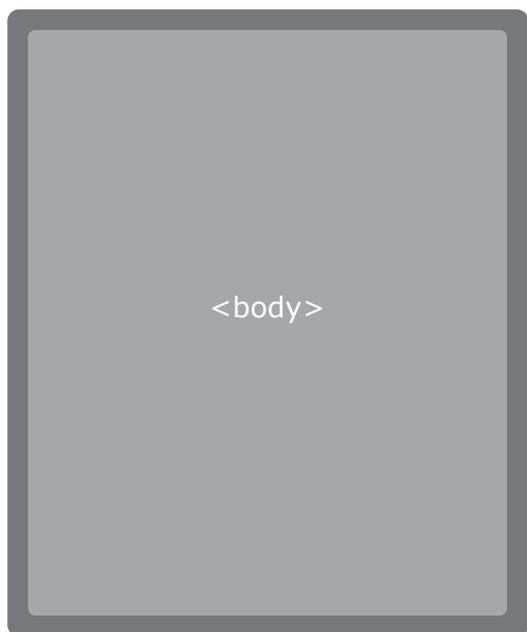
Un contenitore semantico
raggruppa elementi che acquistano un senso nuovo se visti come un tutt'uno.



Arriviamoci con un ragionamento..



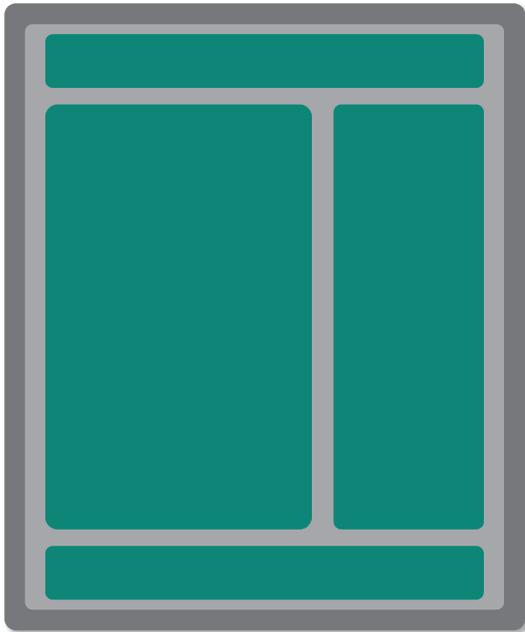
DIV E I NUOVI CONTENITORI SEMANTICI



Tipicamente
una pagina web ha questo aspetto.



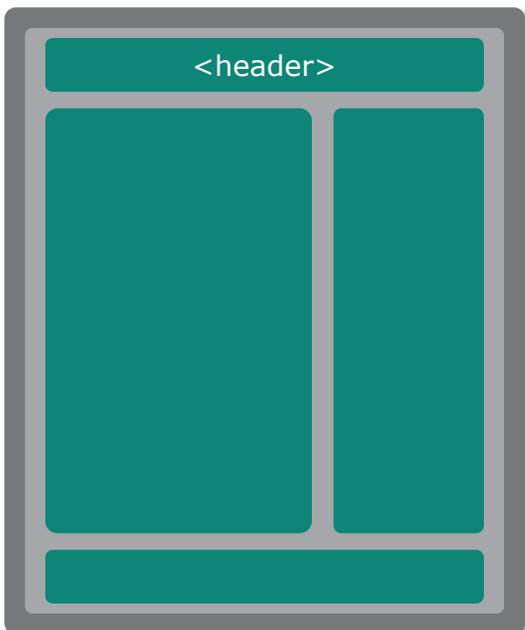
DIV E I NUOVI CONTENITORI SEMANTICI



Tipicamente
una pagina web ha questo aspetto.



DIV E I NUOVI CONTENITORI SEMANTICI

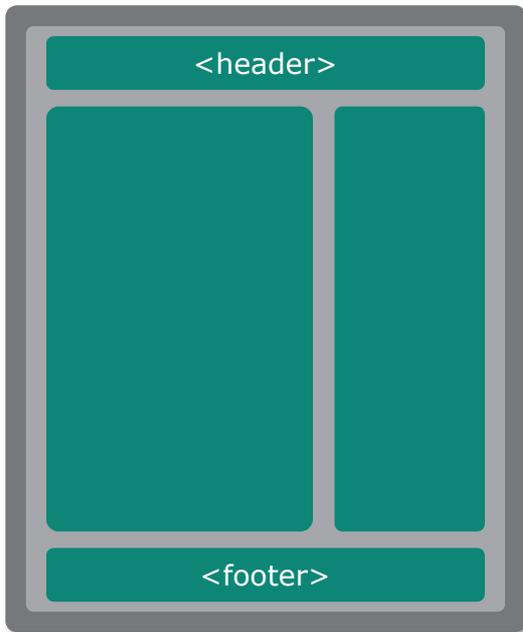


Tipicamente
una pagina web ha questo aspetto.

**header**

Intestazione di una pagina o di un'area specifica. Tipicamente contiene i tag **h1/h6**.

DIV E I NUOVI CONTENITORI SEMANTICI



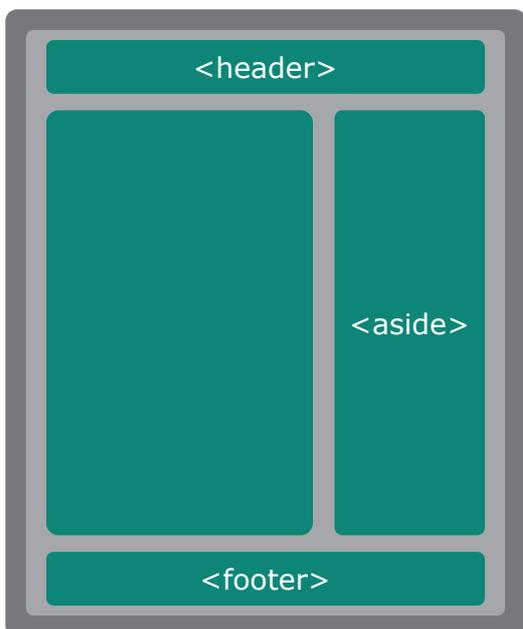
Tipicamente
una pagina web ha questo aspetto.



footer

Ogni pagina (o area) può avere informazioni conclusive quali: autore, link, informazioni di copyright, ecc.

DIV E I NUOVI CONTENITORI SEMANTICI



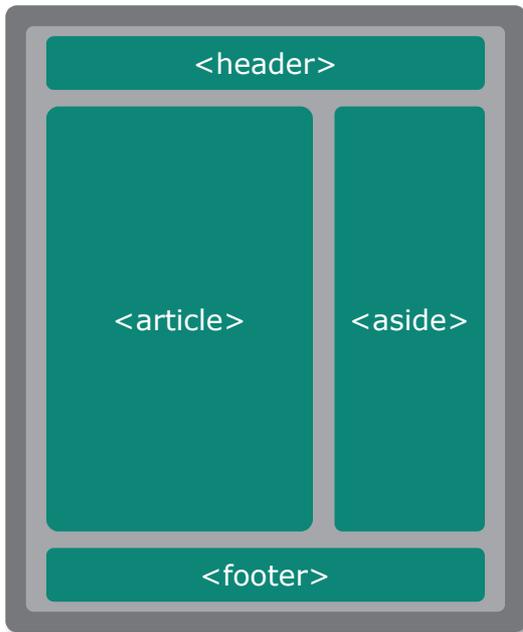
Tipicamente
una pagina web ha questo aspetto.



aside

Definisce un'area laterale il cui contenuto è solo parzialmente correlato al tema principale (menu, link esterni, banner pubblicitari).

DIV E I NUOVI CONTENITORI SEMANTICI



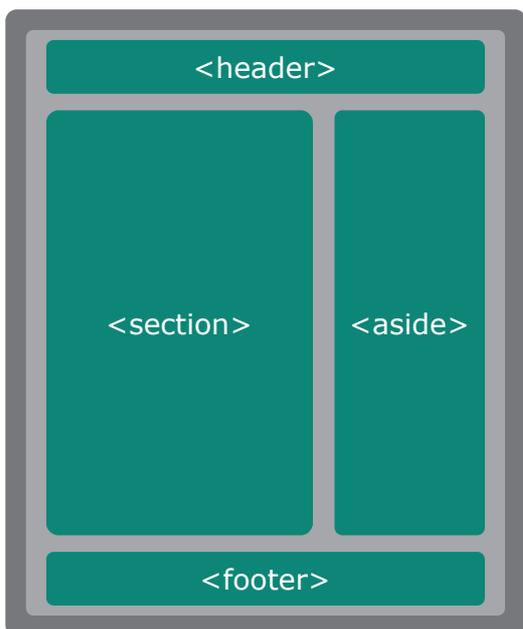
Tipicamente
una pagina web ha questo aspetto.



article

Definisce un'area del sito che contiene un contenuto autonomo e completo che può essere eventualmente ridistribuito.

DIV E I NUOVI CONTENITORI SEMANTICI



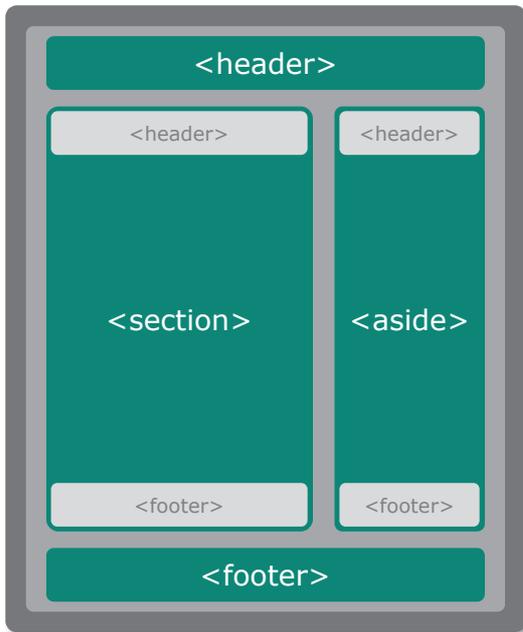
Tipicamente
una pagina web ha questo aspetto.



section

Definisce un'area generica del sito

DIV E I NUOVI CONTENITORI SEMANTICI



Tipicamente
una pagina web ha questo aspetto.

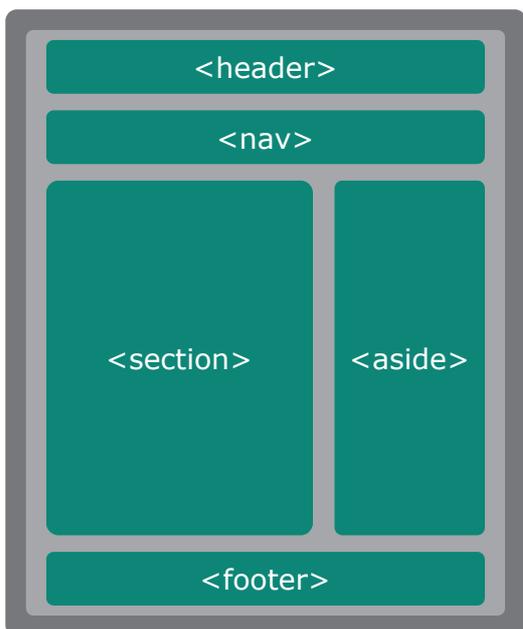


section Definisce un'area
generica del sito

Aree come
aside, **article** e **section**
possono avere **header** e/o **footer**



DIV E I NUOVI CONTENITORI SEMANTICI



Tipicamente
una pagina web ha questo aspetto.



nav Definisce il menu delle
sezioni del sito

MA QUAL È LA DIFFERENZA TRA ARTICLE E SECTION ?

`<article>`

article

Definisce un'area del sito che contiene un contenuto autonomo e completo che può essere eventualmente redistribuito

section

Definisce un'area generica del sito

Facciamo un esempio.

Supponiamo di voler organizzare un post di un blog con tutti i suoi commenti.

MA QUAL È LA DIFFERENZA TRA ARTICLE E SECTION ?

`<header>`

contenuto dell'articolo

`<footer>`

`<section>`

Il post

I commenti

MA QUAL È LA DIFFERENZA TRA ARTICLE E SECTION ?



Un commento 

Il form per
inserire un nuovo
commento 