

Pillole CSS: progettare un sito con font accessibili

Una scelta comune a molti web designer alle prime armi riguarda l'uso dell'unità di misura *pixel* nei fogli di stile. Il pixel è un'unità di misura **relativa al dispositivo** (schermo, palmari, ipad, ecc) che può presentare tuttavia - aldilà della sua semplicità di utilizzo rispetto alle unità percentuali - alcune problematiche dovute alla sua scarsa flessibilità.

L'utilizzo dell'unità pixel in alcuni browser più obsoleti non permette agli utenti di modificare la dimensione del font, e questo ovviamente comporta un grave problema di accessibilità del sito agli utenti ipovedenti che necessitano di ingrandire il carattere.

In altri casi, anche se fosse possibile lavorare sulla dimensione dei vari elementi tipografici, probabilmente tali modifiche comprometterebbero la struttura della pagina rendendo il sito di difficile fruizione.

Come rendere ridimensionabile e accessibile il testo, indipendentemente da browser e dispositivi?

Esistono due tecniche abbastanza semplici da implementare per rendere accessibile in tutti i dispositivi e browser le dimensioni del font.

La prima tecnica consiste nell'usare in combinazione unità assolute e valori di percentuali.

I fogli di stile offrono un'ampia gamma di unità di misura, oltre ai pixel. Noi in questo primo esempio useremo le **unità assolute**, ovvero da *xx-small* a *xx-large*.

Dichiariamo *small* nel font-size di body così:

Inserendo *small* nel body tutti gli elementi della struttura (x)html ereditano questo valore, cosicché avremo una dimensione base di riferimento. *Small* equivale pressochè a 12px. In questo modo possiamo usare nei vari elementi l'unità di misura espressa in percentuale (%), in modo da poter scalare il font in proporzione.

Faccio un esempio per chiarimento, abbiamo questa struttura (x)html;

Avremo così in CSS:

H1 sarà il 150% del valore assegnato al body (*normal*) allo stesso modo con questo meccanismo H2 sarà il 130% dello stesso valore, i paragrafi avranno lo stesso font-size del body, dato che è

100%, (in questo caso uno potrebbe anche omettere la dichiarazione visto che il tag

eredita i valori del body), e le liste avranno un font inferiore a quello del body, cioè l'80%. Un'altra tecnica per avere il testo flessibile, ottima per chi è abituato a "ragionare in pixel", consiste nel combinare valori % con gli **em**.

Oggi la maggior parte dei browser ha un proprio foglio di stile predefinito e i valori di questi fogli di stile sono generalmente abbastanza uniformi tra loro. Il font-size dei browser ha generalmente un valore impostato su *medium* che equivale a 16px, salvo diverse impostazioni modificate dall'utente.

Sulla base di questo concetto, la tecnica sta nell'assegnare al body un font-size di 62,5%:

Perché proprio questo valore?

In questo modo si sfrutta il valore del font-size del foglio di stile predefinito: con il valore 62.5% il font-size si ridimensiona da 16px a 10px, così da avere una dimensione base per il nostro font pari a 10px.

Nel primo caso per aumentare e diminuire si usava la percentuale, con questa tecnica si usano gli em.

Gli em si prestano bene al ridimensionamento font perché sono valori relativi e hanno maggiore flessibilità e precisione.

Aumentare o diminuire gli altri elementi del documento è molto semplice: in questo caso 1em corrisponde al font-size base, cioè a 10px.

Faccio un esempio per dimostrazione:

Gli em saranno dimensionati in base al font-size assegnato al body. Analizzando l'esempio precedente avremo quindi questi valori:

H1 sarà 30px (3em * 10px)

H2 sarà 25px (2.5em * 10px)

H3 sarà 20px (2em * 10px)

UL sarà 8px (0.8em * 10px)

Conclusioni

Entrambi le tecniche sono valide per dare maggior flessibilità al testo del sito, c'è chi preferisce

una o l'altra, io ad esempio uso le keyword assolute in combinazione ai valori percentuali. Spero che questa piccola pillola CSS possa esserti utile nello sviluppo dei tuoi progetti.