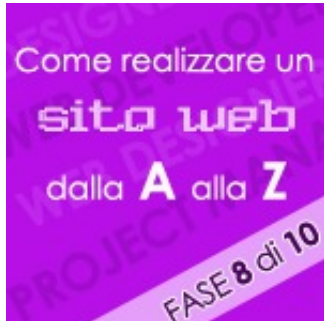


Come trasformare un psd in (x)html e css senza perderci la ragione? Parte 1



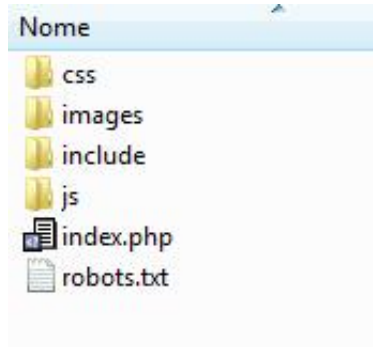
Il nostro compito in questa lezione sarà di **dissezionare** e in seguito di **ricomporre il sito** come in un puzzle della Clementoni, utilizzando i fogli di stile e l'(x)html. È una delle parti più appassionanti del lavoro perché ti permette di seguire lo sviluppo del sito vero e proprio ed è sempre gratificante vedere la tua opera prendere forma mattone dopo mattone. Per uno sviluppatore web può essere una soddisfazione molto simile a quella che può provare un grafico quando concepisce il layout del sito.

Premessa

Iniziamo con il copiare la cartella che utilizziamo come modello di base quando si tratta di sviluppare un nuovo sito web. Sì, perché sarebbe solo uno spreco di tempo dover ricreare ogni volta sempre la stessa struttura di base per il nuovo sito. Se non l'hai ancora fatto allora questa è la volta buona per creare il tuo modello.

Il modello contiene nell'ordine:

- una cartella denominata “**css**” dove sono presenti i fogli di stile di cui si compone il nostro personale Framework CSS (quello che utilizziamo noi è stato generato partendo dalla versione 0.7.1 del framework Blueprint);
- una cartella denominata “**images**” che conterrà tutte le immagini di cui si comporrà il nostro sito;
- una cartella denominata “**include**” in cui collocare eventuali file da includere all'interno delle nostre pagine;
- una cartella denominata “**js**” dove posizionare i file di eventuali script javascript;
- un file denominato “**index.php**” che contiene la struttura standard di un documento web (dichiarazione doctype, head, principali meta tag, collegamento ai fogli di stile, favicon, ecc...);
- un file denominato “**robot.txt**” che conterrà le regole utilizzate dagli *spider* per applicare eventuali restrizioni di analisi sulle pagine del sito.



Passo 1: Come dissezionare correttamente il nostro layout?



Prima di mettere le mani sul codice, è di fondamentale importanza **dissezionare ed esportare in modo corretto le varie sezioni del nostro layout**. Dopo un'accurata analisi ci accorgiamo che nel nostro caso è preferibile suddividere la pagina in tre sezioni:

- **La prima sezione** avrà una larghezza di 1680px, che comprenderà tutta l'estensione dell'immagine in orizzontale, e un'altezza di 632px, indispensabile per visualizzare i palloncini posti lateralmente. Questa immagine sarà il background della sezione principale del sito che chiameremo *wrapper*.
- **La seconda sezione** avrà una larghezza di 1680px e un'altezza di 86px. Questa immagine sarà il pattern assegnato al body che si ripeterà verticalmente sulla base del contenuto testuale inserito nel sito. Ti starai chiedendo: perché proprio 86px? Si tratta di

una dimensione minima che ci permette di replicare l'immagine in verticale [background-repeat:repeat-y;] senza che si noti la ripetizione della stessa.

- **La terza sezione** avrà anch'essa una larghezza di 1680px con un'altezza di 306px. Questa immagine sarà il background della sezione *footer* predisposta alla chiusura del nostro layout. In questa sezione sarà inserita la galleria fotografica.

Una volta definite le varie sezioni, in Photoshop nascondi i livelli superflui, su cui andremo a lavorare in un secondo momento. Nascondi i link laterali, il testo, le miniature della gallery e, come puoi vedere dall'immagine inserita in precedenza, lascia visibile solo le tre sezioni principali.

A questo punto devi **esportare le sezioni**, una per una. Con lo strumento selezione rettangolare, seleziona prima di tutto la prima sezione, stando ben attento alle dimensioni stabilite in precedenza. È un lavoro minuzioso da svolgere con precisione quasi maniacale: spesso anche un solo pixel di differenza può comportare una fastidiosa perdita di tempo e lavoro.

Una volta selezionata la prima sezione, clicca sul menù "*modifica*" alla voce "*copia elementi uniti*". Apri un nuovo documento – che avrà le dimensioni della sezione in memoria già impostate in modo automatico – e incolla la selezione. In questo modo avrai un documento che conterrà solo l'immagine di tuo interesse. Salva l'immagine nella cartella "*images*" e segui lo stesso procedimento con le altre due sezioni del sito.

Passo 2: Mettiamo le mani sul nostro foglio di stile

Apriamo il file "screen.css" che si trova all'interno della cartella css con il nostro editor preferito. La prima sezione che andiamo a creare e che, chiameremo "*wrapper*", **conterrà lo sfondo principale** del nostro sito, con un'altezza minima di 632px, fondamentale per visualizzare i palloncini che sono posti lateralmente.

L'immagine avrà così una dimensione di 1680px di larghezza [risoluzione massima su cui stiamo sviluppando la visualizzazione del sito] per 632px di altezza.

Sotto uno screenshot della stessa:



Per ottenere l'implementazione di min-height anche su browser più datati come internet explorer 6 [che non supporta tale proprietà], ho sfruttato un'altra insufficienza del browser Microsoft. [Qui](#) trovi ulteriori dettagli sulla soluzione adottata.

In questo modo vincoliamo **wrapper** ad avere un'altezza minima di 632px.

Passo 3: Assegniamo un pattern al body

Nell'eventualità in cui il testo della pagina dovesse essere troppo lungo [dopo i 632px di altezza "coperti" dall'immagine che abbiamo assegnato come background a **wrapper**] lo sfondo ritornerebbe a essere vuoto. Per risolvere questo inconveniente sarà necessario quindi **aggiungere un pattern** che dia l'idea di continuità con lo sfondo precedente.

Assegniamo al body l'immagine di sfondo denominata "*pattern.jpg*" di 1680px di larghezza per 86px di altezza.



Passo 4: Iniziamo a scrivere le nostre prime righe di (x)html

Prima di proseguire ulteriormente con il lavoro verifichiamo se quello che abbiamo concepito funziona a livello pratico. Apriamo dunque il file `index.php` con il nostro editor. La prima parte del file conterrà qualcosa del genere:

Aggiungiamo queste poche righe di codice e andiamo a visualizzare il risultato su tutti i browser.



Esattamente quello che volevamo ottenere: come puoi notare il pattern che abbiamo assegnato al tag body si ripete verticalmente andandosi ad integrare perfettamente con l'immagine di sfondo utilizzata nella sezione *wrapper*.

Passo 5: Definiamo il contenitore del sito

Creiamo una sezione che farà da contenitore al nostro sito. Questa sezione, che chiameremo "*container*", avrà una larghezza totale di 960px (920px sarà la dimensione del box al quale assegneremo altri 20px di padding per lato, così da ottenere la larghezza desiderata: 920+20+20) e sarà centrata sullo schermo.

Andiamo a visualizzare il risultato su tutti i browser. Se aggiungessimo un bordo nero a *container*, così da renderlo visibile, avremmo qualcosa come mostrato nell'immagine sotto:



All'interno di *container* andremo ad inserire i contenuti del sito i quali ne regoleranno automaticamente l'altezza.

Passo 6: Dividiamo il contenitore in due colonne

Per raggiungere un buon posizionamento su Google in fase di indicizzazione è importante che il **contenuto testuale del sito venga inserito prima di ogni altra cosa**, ecco perché dobbiamo fare in modo che la colonna di destra, quella che racchiuderà poi il contenuto testuale delle varie pagine, dovrà essere scritta per prima nella nostra pagina html.

Utilizzando questa tecnica css raggiungiamo l'obiettivo che ci siamo prefissati, adesso la colonna di destra (*#right*) potrà essere inserita per prima all'interno della pagina html e poi, a seguire, la colonna di sinistra (*#left*).

Diamo dunque uno sguardo al markup (x)html:

Aggiungendo un bordo alle due colonne e assegnando un'altezza provvisoria di 400px in modo da renderle visibili sullo schermo, possiamo visualizzare un'anteprima dei due box (sui vari browser) che apparirà come nell'immagine mostrata sotto:



Ovviamente i bordi e l'altezza nelle due colonne sono stati impostati solo a scopo didattico, questi poi non saranno presenti nel foglio di stile definitivo.

Passo 7: È il momento di aggiungere il footer

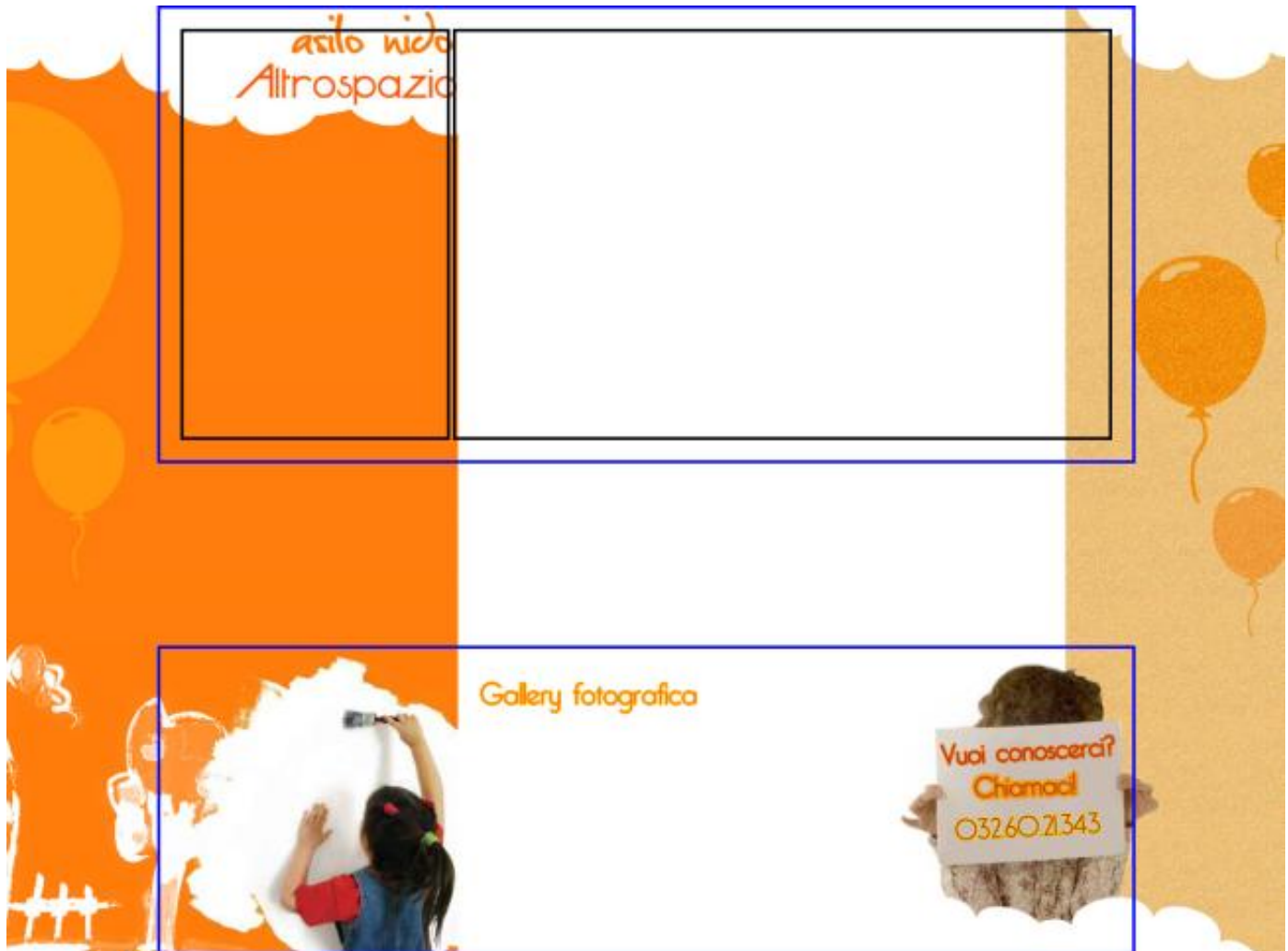
La galleria fotografica che si trova in fondo alla pagina **sarà contenuta in un'altra sezione del sito che chiameremo "footer"**. A questa sezione imposteremo un'altezza pari a 306px come quella dell'immagine di sfondo che gli andremo ad assegnare.



Adesso realizziamo un box che servirà da contenitore alla sezione *footer* in cui andremo dopo a inserire il relativo contenuto: le foto della galleria fotografica.

Il markup (x)html che definisce la struttura del sito che stiamo sviluppando a questo punto del progetto si mostrerà così:

Sotto uno screenshot del lavoro che abbiamo svolto:



Possiamo dire di essere già ad un buon 50% di completamento del lavoro riguardante questa fase di messa in codice del sito. Questa settimana finiamo qui la nostra lezione cosicché tu possa avere la possibilità di approfondire bene la prima parte che è stata trattata. Se dovessi avere qualche dubbio in proposito, c'è la sezione dei commenti per eventuali chiarimenti.

Conclusione

Come hai visto, è di fondamentale importanza **riuscire a dissezionare correttamente il layout che vogliamo ricomporre in codice**. Basta commettere un piccolo errore in questa prima fase e inevitabilmente ci si ritrova a perdere tempo prezioso nello sviluppo di codice (x)html e css che non produrrà i risultati aspettati.

Altro aspetto fondamentale è quello di **verificare, passo dopo passo, la corretta visualizzazione del sito su tutti i principali browser** in modo da correggere subito eventuali imperfezioni e non tralasciare dunque questa fase alla fine della realizzazione dell'intera pagina come spesso può capitare a chi è alle prime armi. Nei casi peggiori potrebbe essere necessario rimettere le mani all'intero codice che si è sviluppato.

La prossima settimana, nella seconda parte di questa lezione, vedrai i passi necessari per disporre i vari contenuti (testata, navigazione, testo, galleria fotografica) all'interno delle relative sezioni che abbiamo appena creato, così da completare il nostro lavoro. Non perdere dunque l'appuntamento.

Gli altri articoli di questa guida:

1. [Come capire le esigenze del cliente al primo incontro?](#)
2. [Come calcolare costi e tempi di consegna di un progetto web evitando di scrivere cifre insensate?](#)
3. [Come evitare futuri ripensamenti da parte del cliente?](#)
4. [Come sviluppare e organizzare la struttura di un sito web?](#)
5. [Come trovare l'ispirazione e disegnare il layout di un sito web?](#)
6. [Come presentare la bozza grafica al cliente?](#)
7. [Come realizzare le pagine interne dopo l'accettazione della bozza?](#)
8. [Come esportare un psd in xhtml e css senza perderci la ragione? \(parte1\) \(parte 2\)](#)
9. [Come indicizzare un sito senza essere degli esperti SEO? \(parte 1\) \(parte 2\)](#)
10. [Come assicurarsi di non aver commesso errori prima di lanciare il sito online?](#)