



Documentation SPIP

Modifier l'habillage graphique

Modifier l'habillage graphique

1. Introduction aux feuilles de style	4
Pourquoi les feuilles de style ?	4
Concrètement.....	5
Notes.....	5
2. Les feuilles de style de SPIP	6
Des styles définis par SPIP	6
Où se trouvent ces définitions de style ?	6
3. Mettez-y votre style !	8
Créez votre feuille de style	8
Respectez la « cascade ».....	9
Des styles qui ont de la « class »	9
La gestion du cache	10
4. Styles des raccourcis typographiques de SPIP	11
Les paragraphes de SPIP	11
Gras et italique.....	11
Les intertitres de SPIP	11
Le trait de séparation horizontal de SPIP	12
Les notes de bas de page	12
5. Styles des liens hypertextes	13
Comment définir l'apparence de ses liens ?	13
Les liens hypertextes de SPIP	13
Pour ne pas s'emmêler les liens	14
Exposer le lien activé	14
6. Styles des citations dans SPIP	15
Les citations de SPIP	15
Poésie.....	15
Le code informatique affiché dans SPIP	15
7. Styles des logos, images et documents	16
Style des logos.....	16
Styles des images et documents	16
Images avec ou sans bordure.....	17
8. Styles des tableaux de SPIP	18
9. Ils sont beaux, mes formulaires !	19
À chaque formulaire son style.....	19
Styles des champs de saisie	20
Des boutons à vos couleurs	20
Organiser ces éléments de façon visiblement logique.....	21
P.-S.	21
10. Ressources CSS pour en savoir plus	22

1. Introduction aux feuilles de style

Avril 2004 — maj : Juillet 2008

Les feuilles de style permettent de centraliser et de gérer de manière beaucoup plus aisée les indications graphiques que l'on insérerait traditionnellement dans le HTML. Elles s'écrivent dans un langage spécifique : le CSS.

Comme vous le savez déjà, SPIP traite séparément le contenu de sa mise en page et son habillage graphique : les squelettes trient et affichent les contenus souhaités en pages HTML, dont l'habillage graphique est réalisé par des feuilles de style CSS. Passez à la vitesse supérieure pour habiller vos squelettes : utilisez les feuilles de style avec SPIP !

Pourquoi les feuilles de style ?

Si vous réalisez des pages Web de manière « traditionnelle », les indications graphiques sont insérées directement dans le code HTML de votre page. Ainsi à chaque fois que vous voulez mettre un texte en rouge, vous écrivez ``. Pour afficher un tableau avec des bordures épaisses, vous écrivez `<table border="2">`.

Avec cette méthode et un site statique (où chaque article a une page HTML spécifique), changer la maquette de tout un site est un cauchemar : il faut rechercher dans tous les fichiers HTML, les portions de code à modifier, et effectuer ces modifications une par une (par exemple remplacer `` par `` si l'on décide que les éléments anciennement affichés en rouge seront désormais en gras).

Comme vous le savez déjà, SPIP améliore beaucoup la situation : vous n'avez plus à modifier des centaines de fichiers HTML, mais juste quelques squelettes ; et votre mise en page est remise à jour automatiquement sur l'ensemble du site. Cependant le problème n'est pas entièrement résolu. Par exemple, mettons que vous ayez décidé d'employer un certain bleu pastel sur beaucoup d'éléments du site, afin de donner une identité graphique à votre site : les liens, les encarts, certains éléments de navigation... sont affichés en bleu pastel. Le jour où vous voudrez remplacer ce bleu pastel par un vert pâle, vous devrez modifier tous les endroits du squelette où ce bleu apparaissait pour le remplacer par le vert pâle. Cela peut être décourageant : il n'est pas aisé dans ces conditions de changer rapidement le rendu des pages, ne serait-ce que pour faire des essais.

La solution réside dans l'utilisation des « **feuilles de style externes** ». Une feuille de style est un fichier où vous définissez un ensemble de propriétés graphiques, et les endroits où elles s'appliquent.

On note deux avantages capitaux des feuilles de style :

- **la feuille de style est un fichier unique et centralisé**, que vous pouvez appliquer à autant de fichiers HTML (et de squelettes SPIP) que vous le désirez ;
- **les propriétés graphiques sont définies une seule fois dans la feuille de style**, quel que soit le nombre d'endroits où ces propriétés sont appliquées dans le HTML.

Concrètement

Pour être appliquée à un fichier HTML (qui peut être un squelette SPIP), la feuille de style doit être déclarée dans l'entête de votre page (entre les balises <head>), de la façon suivante :

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

- Ici le fichier `mes_styles.css` contient les propriétés graphiques que vous voulez appliquer à la page HTML (dans la suite de cette rubrique, on supposera que `mes_styles.css` est le nom que vous avez choisi pour ce fichier).

- Ce fichier porte l'extension « `.css` ». En effet, **CSS** est le nom du langage utilisé pour les feuilles de style, de la même manière que HTML est le nom du langage utilisé pour la réalisation de pages web. **Notez bien que le CSS n'est pas propre à SPIP, il s'agit d'un standard du Web [1].**

Note : une feuille de style peut s'appliquer aussi bien à une page HTML classique (« statique ») qu'à un squelette SPIP (« dynamique »). Cela veut dire que toute astuce CSS valable dans du HTML classique sera aussi utilisable dans un squelette de votre site.

Si vous avez bien lu les paragraphes précédents, vous serez peut-être dubitatifs : oui, il faut apprendre un nouveau langage pour utiliser les feuilles de styles (SPIP n'y est pour rien !). Les CSS n'utilisent pas, en effet, la syntaxe du HTML. Cependant ce langage est très simple, et il suffit de quelques exemples pour mettre le pied à l'étrier. De très nombreuses documentations existent sur ce sujet par ailleurs ; consultez les ressources proposées à cette page : « Ressources CSS pour en savoir plus ».

Notes

[1] La première version de CSS a vu le jour en 1996. C'est un langage de feuille de style, approuvé comme *Recommandation du W3C*.

2. Les feuilles de style de SPIP

Septembre 2001 — maj : Juillet 2008

Le code généré par SPIP est doté de certains styles qu'il convient de définir.

Dans le chapitre précédent, nous avons vu quels étaient les avantages des feuilles de style CSS. Voyons maintenant quel usage spécifique SPIP fait des feuilles de style.

Des styles définis par SPIP

Dans SPIP, certains styles jouent un rôle important : ils servent à modifier les propriétés graphiques des éléments qui ne sont *pas* définis dans *votre* HTML (celui de votre squelette), mais dans le code *généré par SPIP*. En effet, **SPIP associe de lui-même plusieurs styles au code qu'il génère.**

Ainsi, lorsque l'on utilise les raccourcis SPIP dans les articles (permettant par exemple de mettre en gras, en italique, créer des liens hypertextes, des intertitres, des tableaux, etc.), SPIP produit les balises HTML nécessaires à ces effets, chacune de ces balises étant alors dotée d'un sélecteur CSS.

Par exemple, le raccourci suivant :

```
Ceci est un [lien->http://www.uzine.net]
```

est ainsi transformé en code HTML :

```
Ceci est un <a href="http://www.uzine.net" class="spip_out">lien</a>
```

Quel est l'intérêt ? Ces balises portent un nom spécifique dans l'attribut `class` : ce nom définit à quelle « classe » ils appartiennent, c'est-à-dire un ensemble d'éléments HTML qui hériteront des mêmes propriétés graphiques définies dans la feuille de style. Dans notre exemple, le code HTML est complété par le sélecteur CSS intitulé « `spip_out` ». Le webmestre peut donc pousser la personnalisation graphique des liens sortants en modifiant la définition stylistique de « `spip_out` » (couleur différente, fond coloré, police utilisée, etc.).

L'apparence de la plupart des raccourcis SPIP peut ainsi être paramétrée dans les feuilles de style. Cela vaut aussi pour les formulaires automatiques (répondre à un forum, signer une pétition...) et d'autres encore. Certains de ces styles sont très utiles, voire indispensables, d'autres seront réservés aux webmestres qui souhaitent obtenir des effets exotiques.

Où se trouvent ces définitions de style ?

Les propriétés graphiques appliquées aux pages HTML sont regroupées dans les fichiers `.css` qui accompagnent les squelettes. Les squelettes par défaut et leurs feuilles de style sont regroupés dans le répertoire *squelettes-dist*.

Les définitions de style propres à SPIP se trouvent dans la « feuille de style externe » nommée `spip_style.css`. Celle-ci regroupe les définitions des styles associées au code généré par SPIP.

Une autre feuille de style, `spip_admin.css`, permet de contrôler l'apparence des boutons d'administration (« recalculer cette page », etc.).

Vous pouvez les modifier (c'est même conseillé : « Mettez-y votre style ! »), mais notez bien que vous ne pouvez pas les renommer. **Ces styles sont indispensables et doivent nécessairement être définis pour un bon affichage de vos squelettes.**

3. Mettez-y votre style !

Avril 2004 — maj : octobre 2010

Vous pouvez modifier les styles fournis avec SPIP et ajouter les vôtres, en créant votre propre feuille de style. Voici comment.

Si vous connaissez le langage CSS, vous pouvez très facilement modifier l'apparence de votre site, sans même avoir besoin de connaître le langage des boucles et balises de SPIP.

Créez votre feuille de style

Lors de l'installation de SPIP, les squelettes sont distribués avec plusieurs « feuilles de style externes » qui regroupent les indications produisant l'habillage graphique du site. Vous pouvez modifier ces fichiers et ajouter vos propres définitions de style, mais il est préférable de le faire dans votre propre fichier CSS afin de pas voir vos ajouts « écrasés » lorsque vous installerez une nouvelle version de SPIP.

Important : ne travaillez jamais directement dans les fichiers fournis par défaut, sinon vous risqueriez de perdre toutes vos modifications à chaque mise à jour de SPIP ! Pour éviter cela, faites une copie des fichiers que vous souhaitez modifier.

1. Créez un fichier `mes_styles.css` (ou tout autre nom que vous avez décidé de lui donner) et rangez-le dans votre dossier « squelettes ». Vous copierez dans ce fichier les définitions de styles que vous souhaitez utiliser et modifier ; mais pour la suite de ce tutorial, nous allons considérer que vous partez d'une feuille vierge.

2. Appelez cette feuille de style dans l'entête de votre squelette, c'est-à-dire entre les balises `<head>` du fichier HTML (aux côtés du `title` et autres `meta`). De la façon suivante :

```
<link rel="stylesheet" type="text/css" href="#CHEMIN{mes_styles.css}" />
```

Bien souvent une seule feuille de style suffit pour tout l'habillage graphique d'un site, mais vous pouvez déclarer de cette façon autant de feuilles de style que nécessaire.

- SPIP vous facilite la vie : perso.css

Pour découvrir un peu la personnalisation des styles, vous pouvez aussi tout simplement créer un fichier `perso.css` dans le dossier `squelettes/`.

Comme SPIP charge ce fichier en dernier dans le head, les styles qui y sont définis ou redéfinis deviendront prioritaires. L'utilisation de ce fichier est bien pratique pour :

- modifier l'un ou l'autre style
- tester des modifications
- déclarer des styles perso

Pour des développements et personnalisations plus conséquentes, il reste conseillé d'utiliser vos propres feuilles de style déclarées comme indiqué ci-dessus.

Respectez la « cascade »

Il est important de garder à l'esprit le fonctionnement « **en cascade** » du CSS (*Cascading Style Sheets* signifie littéralement « feuilles de style en cascade ») : lorsque plusieurs définitions de style concernent un même élément, **est appliqué en priorité le style le plus proche de l'élément**. L'ordre dans lequel les styles sont « lus » a donc une importance.

- Feuilles de style externes

Si vous utilisez plusieurs feuilles de style, notez que l'ordre dans lequel celles-ci sont appelées dans l'entête de la page a une importance. Si vous appelez *d'abord* `mes_styles.css` et *ensuite* `spip_style.css` : ce sont les styles de cette dernière, plus *proches*, qui s'appliqueront prioritairement aux vôtres. Pensez donc à faire l'inverse :

```
<link rel="stylesheet" type="text/css" href="#CHEMIN{spip_style.css}">
<link rel="stylesheet" type="text/css" href="#CHEMIN{mes_styles.css}">
```

- Styles définis dans le code HTML

Si vous ne souhaitez pas toucher aux fichiers CSS, vous pouvez continuer à insérer, par endroits, des indications graphiques directement dans le code HTML de vos squelettes : en définissant quelques styles dans le `head`, et/ou en plaçant des indications de style directement dans les balises HTML de la page.

Les styles placés directement dans les balises, étant au plus *proches* des éléments concernés, seront prioritaires sur ceux définis dans le `head`, eux-mêmes prioritaires sur ceux des feuilles de style *externes*.

Des styles qui ont de la « class »

Comment fait-on alors pour changer, par exemple, l'apparence de tous les intertitres SPIP ? C'est très simple. Ouvrez votre fichier `mes_styles.css` dans un éditeur de texte et ajoutez-y la ligne suivante :

```
h3.spip { color: red; font-size: 18px; }
```

Rechargez la page : tous les intertitres apparaissent comme par magie en rouge ; remarquez de plus que les autres `h3` de votre page, s'il y en a, ne sont *pas* affichés en rouge.

Expliquons brièvement la syntaxe de cette règle de mise en page :

- `h3.spip` juste avant les accolades signifie que la règle ne s'applique qu'aux `<h3>` dotés d'un attribut `class` égal à « `spip` ».

Notez bien : ni les `<h3>` n'ayant pas cet attribut, ni les balises ayant cet attribut sans être des `<h3>`, ne seront concernés.

Si vous ajoutez vos propres styles, sachez que la valeur donnée à l'attribut `class` est totalement arbitraire. La seule chose qui compte, est que vous utilisiez bien le même nom dans le code HTML (`class="toto"`) et dans votre feuille de style CSS (`.toto { ... }`). Rappelons toutefois que vous ne pouvez pas renommer les `class` associées au code généré par SPIP (dont les définitions de style sont regroupées dans `spip_style.css`).

- Les accolades contiennent la liste des propriétés graphiques associées au style ainsi défini. Ici nous voyons que la couleur est réglée à rouge et que la police de caractères doit être affichée avec une taille de 18 pixels.

Notons que toutes les propriétés non définies dans cette liste garderont leur valeur habituelle pour la balise considérée ; dans le cas présent, le h3 générera toujours un texte en gras, car rien dans cette définition de style ne dit le contraire.

La gestion du cache

Le fait que les styles soient définis dans un fichier séparé a une conséquence importante. En effet, ce fichier, au contraire de vos squelettes n'est pas géré par SPIP (il n'en a pas besoin !). Cela signifie que **si vous modifiez une feuille de style, vous n'avez pas besoin de vider le cache de SPIP : il suffit de recharger la page dans votre navigateur**. Cela rend le réglage de la mise en page encore plus aisé.

Rappelons tout de même que votre feuille de style doit être déclarée dans vos fichiers HTML, et que ceux-ci doivent être recalculés une première fois pour que cette déclaration soit prise en compte.

4. Styles des raccourcis typographiques de SPIP

Juin 2006 — maj : Mai 2009

Voici les styles CSS associés aux raccourcis typographiques les plus courants et les plus simples de SPIP.

Pour faciliter la mise en forme des textes, SPIP propose un certain nombre de « raccourcis SPIP » qui font l'objet d'un traitement automatisé : ils sont remplacés par le code HTML correspondant, doté d'un style (généralement la classe `spip`). Pour affiner encore davantage la mise en forme des textes, vous pouvez modifier les styles CSS associés à ces raccourcis typographiques.

Les paragraphes de SPIP

Pour créer des paragraphes dans SPIP, il suffit de laisser une ligne vide. Les paragraphes ainsi générés par SPIP sont dotés de la classe `spip` :

```
<p class="spip">Voici le texte de mon paragraphe</p>
```

Pour ajuster, par exemple, l'espace entre chaque paragraphe, modifiez la définition de style : `p.spip`.

Remarque : ceci n'est valable que pour les textes en plusieurs paragraphes.

Gras et italique

Le gras et l'italique sont générés par les raccourcis suivants :

```
Du texte {en italique}, du texte {{en gras}}
```

Leur apparence est contrôlée par les définitions de style respectives : `i.spip` et `strong.spip`. Ces styles sont peu utiles, et ne sont d'ailleurs pas définis par défaut.

Les intertitres de SPIP

Les intertitres sont créés par le raccourci suivant :

```
{{{Un intertitre}}}
```

Leur apparence est contrôlée par la définition de style `h3.spip`. Ce style est sans doute l'un des plus importants, car il permet de définir la taille, la police et le positionnement des intertitres dans les articles : vous serez certainement amenés à le modifier en fonction de vos choix graphiques et typographiques.

Notez en particulier les attributs `margin` et `padding` qui permettent d'agir sur l'espacement de l'intertitre avec les paragraphes précédent et suivant. Sans ce réglage, l'intertitre serait soit trop « collé » au reste du texte, soit trop espacé.

Le trait de séparation horizontal de SPIP

Le trait de séparation horizontal est généré par ce raccourci SPIP :

Son apparence est contrôlée par la définition de style : `hr.spip`.

Remarque : La ligne horizontale (`<hr />`) est un élément délicat à styler, car les styles CSS qui lui sont appliqués sont interprétés différemment selon les navigateurs.

Les notes de bas de page

Les notes de bas de page, sont créées par le raccourci :

Le texte[[Ceci est une note de bas de page.]].

Leur apparence est contrôlée par plusieurs définitions de style :

- `a.spip_note` permet de différencier visuellement les appels des notes de bas de page des autres liens (c'est-à-dire leurs numéros, tant dans le corps du texte qu'en bas de page) ;
- `p.spip_note` contrôle l'affichage des notes elles-mêmes ; bien souvent inutile (on préférera en effet définir ce style plus simplement, à partir du style de la balise parente.).

5. Styles des liens hypertextes

Juin 2006 — maj : Juillet 2008

Comment définir l'apparence de ses liens ?

Sur le web, les liens hypertextes sont traditionnellement caractérisés par la couleur bleue et le soulignement. Mais vous avez certainement remarqué que cette présentation varie d'un site à l'autre. Vous pouvez très facilement personnaliser l'apparence de vos liens avec les styles CSS. Exemple très classique :

```
a { color: green; text-decoration: none; }
a:hover { color: red; text-decoration: underline; }
```

- « **a** » concerne tous les liens affichés sur votre page web, c'est-à-dire toutes les balises <a>, sans exception, qu'elles aient ou non un attribut class. Dans l'exemple ci-dessus, les liens s'afficheront en vert, sans « décoration » : ils ne sont pas soulignés.

- Plusieurs états sont disponibles pour les liens avec les « pseudo classes » (:hover, :visited, etc.). Ainsi « **a:hover** » concerne les liens « survolés ». Dans l'exemple ci-dessus, les liens deviennent rouges et soulignés lorsqu'ils sont survolés par le pointeur de la souris.

Notez que la recommandation CSS2 précise que, pour être pleinement prise en compte, la règle a:hover doit être placée après les autres.

Les liens hypertextes de SPIP

SPIP va plus loin en vous permettant de différencier graphiquement les différents types de liens, notamment les liens internes au site et les liens vers d'autres sites. Cela se fait avec quelques définitions de style spécifiques, que nous vous recommandons de personnaliser à votre goût :

- **a.spip_in** concerne les liens à l'intérieur de votre propre site. Par exemple : Le raccourci [->article1177] génère un lien interne, vers l'article 1177 de votre site, ainsi : [Les feuilles de style de SPIP](#)

- **a.spip_out** concerne les liens vers l'extérieur de votre site. Par exemple : Le raccourci [uzine->http://www.uzine.net] affiche le lien externe suivant : [uZine](http://www.uzine.net)

- **a.spip_url** traite les adresses URL transformées en lien hypertexte. Par exemple : Le raccourci [->http://www.uzine.net] affiche directement l'URL, avec un lien hypertexte vers cette adresse, ainsi : <http://www.uzine.net>

- **a.spip_glossaire** concerne les liens vers le glossaire externe (en l'occurrence l'encyclopédie en ligne Wikipédia).

Par exemple : Le raccourci [?SPIP] génère le lien suivant : [SPIP](#)

Pour ne pas s’emmêler les liens

Notez bien que, pour un lien donné, plusieurs définitions de style interviennent. Par exemple, si vous avez précisé :

```
a { color: green; text-decoration: underline; }
a.spip_in { color: orange; }
```

Les liens internes (dotés de la classe `spip_in`) sont de couleur orange, mais héritent également du soulignement appliqué à `<a>`. Les autres liens, y compris ceux qui ne sont pas générés par SPIP, s’afficheront en vert souligné.

On note ici une propriété fondamentale des feuilles de style : les règles graphiques s’appliquent dans l’ordre allant de la plus générique à la plus spécifique. Cela permet de spécifier un comportement général pour la plupart des éléments, et de modifier ce comportement pour un plus petit sous-ensemble d’éléments. Cette caractéristique fait toute la puissance des feuilles de style.

Exposer le lien activé

On peut mettre en évidence, dans une liste de liens, celui concernant la page où l’on se trouve. Le style que l’on utilisera pour ce faire est : `.on`. Voir : « Exposer un article dans une liste »

6. Styles des citations dans SPIP

Juin 2006 — maj : Janvier 2007

Citations d’auteurs, de code informatique ou de poésie, SPIP facilite l’insertion de citations dans vos textes. *Code is poetry !*

Les citations de SPIP

Pour simplifier l’insertion de citations dans vos textes, SPIP propose le raccourci suivant :

```
<quote>Ceci est une citation.</quote>
```

ce qui donne :

Ceci est une citation.

Son apparence est contrôlée par la définition de style `blockquote.spip`.

Signalons l’existence d’une autre balise HTML, que vous pouvez également utiliser dans vos textes : à la différence de celle utilisée dans le raccourci de SPIP (`<blockquote>`) qui « va à la ligne », `<q>` permet de faire une citation au fil du texte.

Poésie

SPIP propose de plus une mise en forme particulière pour la poésie, avec le raccourci suivant :

```
<poesie>Voici de la poésie.</poesie>
```

ce qui donne :

Voici de la poésie.

Son apparence est contrôlée par la définition de style `div.spip_poesie`.

Le code informatique affiché dans SPIP

Pour citer du code informatique au fil du texte, on utilise le raccourci SPIP suivant :

```
<code>Du code dans le texte</code>
```

Le style associé est `.spip_code`.

Le très astucieux raccourci `<cadre>...</cadre>` permet de présenter du code informatique dans un bloc de formulaire, ce qui facilite le copier-coller du code. Son apparence est contrôlée par la définition de style : `.spip_cadre`.

Ces deux derniers raccourcis, et leurs définitions de style correspondantes, sont peu utilisés, sauf dans le cas d’une documentation technique (comme celle-ci) où l’on doit citer des morceaux de code informatique, des noms de fichiers ou de répertoires.

7. Styles des logos, images et documents

Juin 2006 — maj : Décembre 2007

Les styles CSS associés par SPIP aux logos, aux images et autres documents permettent de contrôler leur affichage dans la page.

Style des logos

Les logos des éléments (rubriques, articles, brèves, auteurs, sites) sont systématiquement dotés du style `.spip_logos`, placé sur la balise HTML ``. Ce style peut être très utile pour, par exemple, déterminer la position du logo.

Ainsi pour aligner à droite les logos placés dans le cartouche, par exemple, tout en ménageant un espace entre l'image et le texte, vous ferez simplement :

```
.cartouche .spip_logos {
    float: right;
    margin-left: 1em;
}
```

Styles des images et documents

L'insertion avec les raccourcis `<docXX|left>` et `<imgXX|right>`, de documents et images dans le corps de texte d'un article (ou d'une brève), est contrôlée par des styles CSS.

Trois définitions de style permettent de personnaliser l'affichage des documents, de leurs titres et légendes :

- `.spip_documents` concerne la boîte qui contient la vignette et les informations du document (`<docXX|left>`) ou l'image insérée sans titre ni descriptif (`<imgXX|right>`);
- `.spip_doc_titre` contrôle l'affichage du titre du document;
- `.spip_doc_descriptif` contrôle l'affichage du descriptif du document.

Trois styles sont utilisés en complément, indispensables pour définir le positionnement du document ou de l'image dans la page :

- `.spip_documents_center` quand le document est centré (`<docXX|center>`);
- `.spip_documents_left` quand le document est aligné à gauche (`<docXX|left>`);
- `.spip_documents_right` quand le document est aligné à droite (`<docXX|right>`).

Notez bien que ces styles concernent de la même façon les images (`<imgXX>`) et les documents (`<docXX>`).

NB : Avant de modifier radicalement ces définitions de style, notez que celles-ci sont également utilisées, dans les squelettes par défaut (depuis [SPIP 1.9](#)), pour styler les éléments du portfolio et des listes de documents joints (aux articles et/ou rubriques).

Images avec ou sans bordure

Il est possible d'appliquer une bordure aux images. Par exemple : `.spip_documents img { border: 1px solid #666; }` encadrera vos images d'une fine bordure grise.

Inversement, pour ne pas voir vos logos encadrés de cette horrible bordure bleue qui signale traditionnellement les « images cliquables » sur certains navigateurs (comme FireFox), n'oubliez pas de régler leur bordure à zéro : `.spip_logos { border: 0; }`.

8. Styles des tableaux de SPIP

Juin 2006 — maj : Juillet 2008

Les styles CSS permettent de paramétrer finement l’affichage des tableaux générés par SPIP.

Les tableaux constituent un moyen facile de présenter l’information en rangées et colonnes de cellules.

Pour tirer profit de cet article, mieux vaut connaître la syntaxe HTML des tableaux. La balise `<table>`, qui permet d’insérer un tableau dans la page, est l’une des balises les plus utilisées en HTML.

Les tableaux sont créés dans SPIP de la façon suivante :

```
||Légende du Tableau|Résumé du Tableau||
| {{Nom}} | {{Date de naissance}} | {{Ville}} |
| Jacques | 5/10/1970 | Paris |
| Claire | 12/2/1975 | Belfort |
| Martin | 1/31/1957 | Nice |
| Marie | 23/12/1948 | Perpignan |
```

ce qui produit cet affichage :

Légende du Tableau

Nom	Date de naissance	Ville
Jacques	5/10/1970	Paris
Claire	12/2/1975	Belfort
Martin	1/31/1957	Nice
Marie	23/12/1948	Perpignan

Les styles permettent de paramétrer finement l’affichage des tableaux :

- `table.spip` permet de modifier le comportement général du tableau, notamment ses dimensions, ses marges et sa position (calé à gauche, centré, etc.) ;
- `table.spip caption` concerne la légende (optionnelle) du tableau ;
- `table.spip tr.row_first` concerne la « première ligne » du tableau (ici en jaune). Pour que la première ligne soit prise en compte comme rangée de titres, il faut que chacun des éléments qu’elle contient soient en gras ;
- `table.spip tr.row_odd` et `table.spip tr.row_even` pour les autres lignes. Un des intérêts de ces styles est la possibilité d’appliquer deux couleurs différentes via « `row_odd` » et « `row_even` » (ici, gris clair et gris foncé), permettant d’alterner les couleurs d’une ligne à l’autre, ce qui facilite agréablement la lecture du tableau.
- `table.spip th` et `table.spip td` concernent les cellules du tableau, et permettent, par exemple, de contrôler leur espacement intérieur (`padding`), afin d’aérer la présentation.

9. Ils sont beaux, mes formulaires !

Avril 2004 — maj : Juillet 2008

Vous avez personnalisé la mise en page et la typographie de votre site, mais maintenant ce sont les formulaires SPIP qui jurent totalement sur le reste ! Pas de panique, là aussi les feuilles de style remédient au problème.

Différents formulaires sont utilisés dans le site public, pour le moteur de recherche interne, la rédaction des messages des forums, les inscriptions à l'espace privé, etc. Il en va pour ces formulaires SPIP comme pour le reste : leur aspect graphique peut être modifié via CSS afin de s'insérer sans hiatus dans votre design.

Pour tirer profit de cet article, mieux vaut connaître les balises HTML propres aux formulaires.

À chaque formulaire son style

Tous les formulaires SPIP utilisés dans le site public sont contenus dans une `div` dotée du même style, `.formulaire_spip`, qui permet d'appliquer facilement une modification devant concerner l'ensemble de ces formulaires.

Par exemple, pour mettre en gras tous les descriptifs des champs de saisie (`<label>`) de vos formulaires, vous stylerez ainsi :

```
.formulaire_spip label {
    font-weight: bold;
}
```

Chaque formulaire est, de plus, doté d'un style qui lui est propre. Celui-ci permet, inversement, de modifier l'apparence de certains formulaires en particulier, sans interférer sur les autres.

Chacun de ces styles est nommé de même que la balise appelant le formulaire et son squelette. Par exemple, le fichier HTML du formulaire de recherche est `formulaire_recherche.html` ; celui-ci est inséré dans les squelettes grâce à la balise `#FORMULAIRE_RECHERCHE` ; et le style qui lui est associé est donc `.formulaire_recherche`.

Les squelettes des formulaires sont déplacés et ainsi renommés :

```
dist/formulaires/recherche.html.
```

Styles des champs de saisie

Le style `.form1` est appliqué aux champs de saisie des formulaires. Il permet de définir la couleur du fond et la largeur des champs de saisie, mais aussi leur taille et police de caractère. Par exemple :

```
.form1 {
  width: 99%;
  padding: 1px;
  border: 1px solid #666;
  font-family: Verdana;
  font-size: 11px;
}
```

Ce style est particulièrement utile pour homogénéiser l'ensemble des champs de saisie, quelques soient les balises qui le reçoivent, telles que `<input>` et `<textarea>`, toutes deux présentes dans le formulaire de forum.

Chaque champ de saisie est étiqueté d'un terme explicatif, enveloppé dans une balise HTML `<label>`. On modifiera l'apparence de ces étiquettes par cette définition de style : `.formulaire_spip label`.

Les styles CSS permettent non seulement de changer les couleurs et les polices de caractères, mais aussi de gérer le positionnement relatif des objets dans la page. Ils permettent même de définir précisément la disposition des éléments entre eux (par exemple : `<input>`, `<textarea>` et `<label>`), sans utiliser de `<table>` pour la mise en page.

Des boutons à vos couleurs

Une nouvelle qui ravira les débutant-e-s en CSS : on peut changer la couleur des champs mais aussi des boutons des formulaires¹. Le style `.spip_bouton` est celui utilisé pour les boutons des formulaires SPIP.

Par exemple, pour que les boutons aient un fond bleu clair, et une bordure suffisamment visible (épaisse, en relief et bleue foncée), modifiez la règle suivante dans votre feuille de style :

```
.spip_bouton {
  background-color: #b0d0FF;
  border: 2px outset #000060;
  color: black;
}
```

Le formulaire de forum propose une barre de raccourcis typographiques. L'apparence de celle-ci est contrôlée par le style `.spip_barre`. Ainsi, pour modifier l'apparence des icônes qui la composent, et, par exemple, distinguer celles-ci au survol de la souris, vous définirez les styles pour `.spip_barre a img` puis `.spip_barre a:hover img`.

¹ Notons cependant que certains navigateurs imposent leurs propres boutons stylisés et ne vous laisseront pas en changer l'aspect.

Organiser ces éléments de façon visiblement logique

Les différents éléments d'un formulaire sont plus rigoureusement regroupés en « blocs logiques » grâce aux balises HTML dédiées `<fieldset>` et `<legend>`. Leur apparence est donc contrôlée par `.formulaire_spip fieldset` et `.formulaire_spip legend`, tout simplement (ceci était autrefois réalisé avec `.spip_encadrer`). Utile pour encadrer chaque partie d'une bordure, et aérer vos formulaires en espaçant ces blocs les uns des autres, par exemple.

Quelques styles complémentaires permettent d'affiner encore la présentation de vos formulaires :

- L'apparence des messages d'erreur et autres réponses renvoyées par les formulaires peut être personnalisée par le style `.reponse_formulaire`.
- Certains formulaires permettent de prévisualiser les informations saisies avant de les envoyer. L'apparence de cette prévisualisation est contrôlée par le style `.previsu`.
- Enfin, ce dernier style n'est pas utilisé dans les formulaires, mais est lié au formulaire de recherche interne au site : `.spip_surligne` permet de mettre en évidence les termes recherchés dans les pages de résultats.

P.-S.

Cet article de Pompage.net introduit au style des formulaires :

« <http://pompage.net/pompe/cssdezero-13/> ».

Lire aussi :

<http://blog.alsacreations.com/2007/06/12/366-comment-ne-pas-styler-les-elements-de-formulaire>

10. Ressources CSS pour en savoir plus

Avril 2004 — maj : Décembre 2007

Le CSS n'est pas propre à SPIP. C'est un langage normalisé par le W3C et très utilisé sur le Web pour l'habillage graphique. Vous êtes libres de vous contenter du niveau abordé dans ces pages, mais nous vous invitons à consulter la très nombreuse documentation existant sur le CSS.

Citons quelques ressources intéressantes.

Ressources francophones sur le CSS

- **OpenWeb** (<http://openweb.eu.org/>) est un site offrant à la fois un regard expert sur le Web et des exemples concrets d'utilisation sur les CSS et autres standards du Web.
- un cours « CSS débutant » (<http://www.mammothland.net/cours/css/>), de Mammothland ;
- un excellent tutorial : « **CSS : on reprend tout à zéro !** » (<http://pompage.net/pompe/cssdezero-1>), par pompage.net, qui permet de débiter en douceur et en toute beauté ;
- une collection de recettes (<http://css.alsacreations.com/>) pour une utilisation efficace des CSS, et un forum CSS et standards W3C (<http://forum.alsacreations.com/>) pour s'entre-aider, chez Alsacreations ;
- un pense-bête (<http://fr.selfhtml.org/navigation/css.htm>), pour ne pas se perdre dans le feu de l'action ;
- et surtout, la traduction française de la documentation officielle, assez aride mais incontournable : les **spécifications CSS2** (<http://www.yoyodesign.org/doc/w3c/css2/>) du W3C.

Ressources anglophones

- Listes d'éléments : des exemples d'effets graphiques (<http://css.maxdesign.com.au/index.htm>) à foison ;
- de très bons tutoriaux (<http://css.maxdesign.com.au/floatutorial/>) sur le positionnement d'objets avec les CSS ;
- le W3C a ses propres *tips'n'tricks...* (<http://www.w3.org/Style/Examples/007/>)
- et pour ceux qui ont le goût du risque, l'intégrale des spécifications originales du W3C ! (<http://www.w3.org/Style/CSS/>)

Petits outils bien utiles

Si vous utilisez l'excellent navigateur Firefox, des extensions telles que *CSS Viewer*, le plug-in *EditCSS*, ou la barre d'outils *Web Developer* vous rendront des services appréciables en matière de CSS.