

Formation à spip en 2h



Spip est un logiciel libre de type “gestion de contenu”. Il permet de créer et de gérer facilement un site internet en séparant le contenu du site et la présentation à l'écran.

Programme du cours

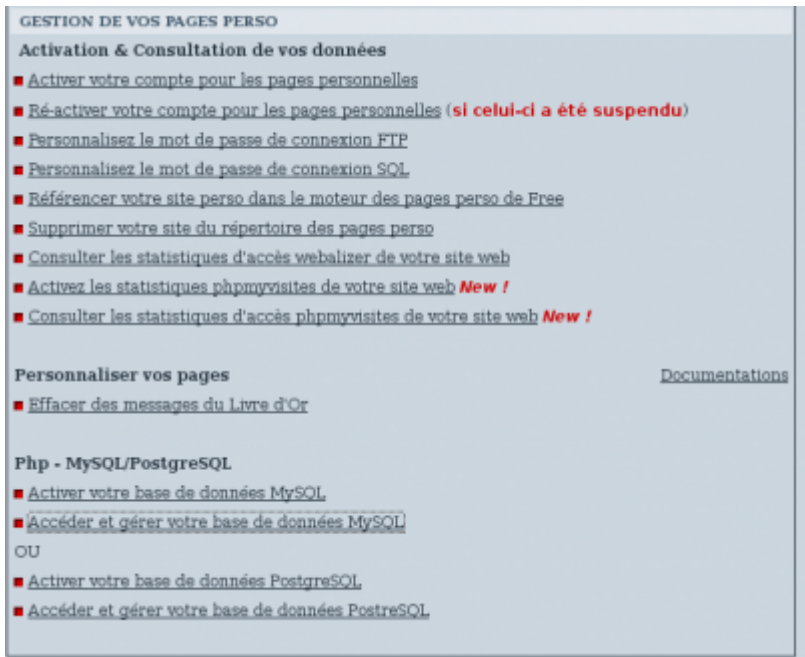
- 20 minutes : installation de spip sur un hébergement gratuit type free.fr (fireftp, spip_loader, mysql) et découverte de l'interface privé.
- 20 minutes : spip, comment ça marche ? (les squelettes, la base de données, le cache)
- 40 minutes : personnalisation des squelettes : “des boucles et des balises...”
- 40 minutes : les bases de xhtml 1.0 et css 2

Installation de spip chez free.fr

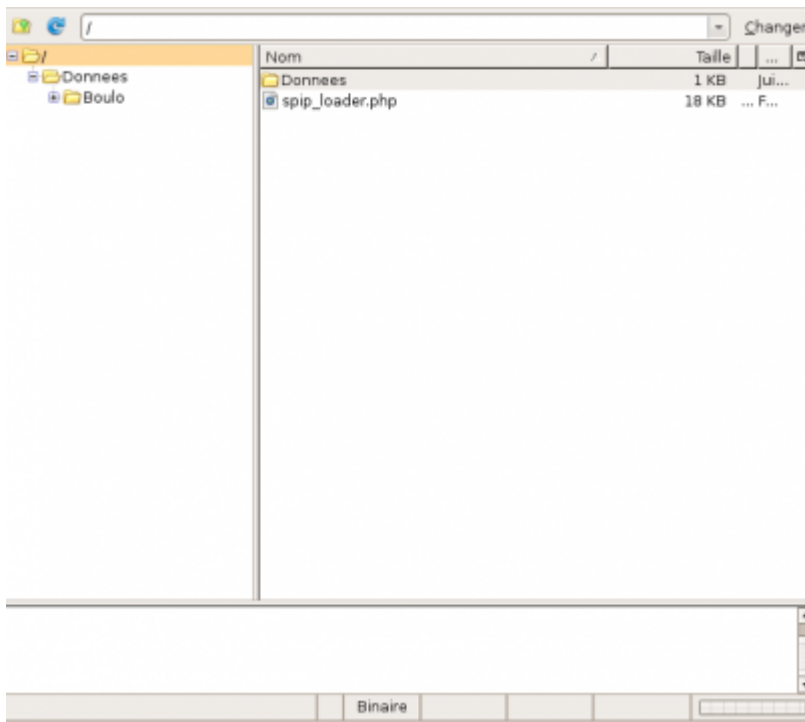
- Aller sur le site www.free.fr, rubrique "accès gratuit".
- Complétez les informations. Vous recevrez vos identifiants de connexion dans les 10 jours chez vous.

5 à 10 jours plus tard ...

- Rendez-vous sur votre gestionnaire sur free.fr et activez votre base de données Mysql.



- Lancez votre navigateur préféré (firefox) et installez l'extension [fireftp](#).
- Configurez le serveur : ftpperso.free.fr, login : `le_votre`, mot de passe : `le_votre`
- Téléchargez le fichier [spip_loader](#) depuis le site officiel de spip.
- Déposez le à la racine de votre dossier `www` sur le serveur de free.

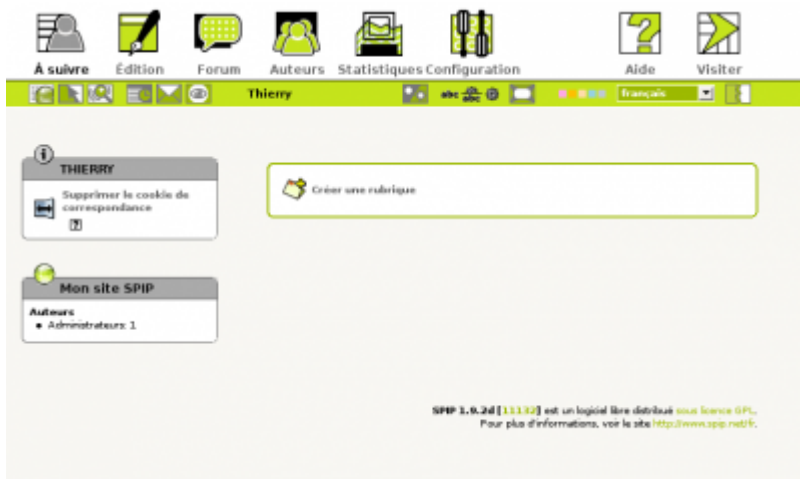


- rendez vous à l'adresse http://votre_login.free.fr/spip_loader.php et suivez les indications. Après quelques clics et quelques saisies, votre site est prêt à fonctionner.
- Si vous rencontrez des difficultés avec votre base de données, vous pouvez la gérer à l'adresse <http://sql.free.fr/phpMyAdmin/>

Découvrir l'interface privé de spip

- Rendez-vous à l'adresse http://votre_login.free.fr/ecrire et saisissez les logins et mots de passe créés à l'installation.

- Créez une rubrique à la racine du site et un premier article dans cette rubrique.



- Allez visiter votre site Ça y est, vous êtes en ligne !

Spip, comment ça marche ?

Spip sépare le contenu de votre site (les textes, les photos, les documents) et la mise en page. Le même contenu peut ainsi être présenté de manières totalement différentes en changeant les "squelettes".

Un squelette est un fichier décrivant une page de votre site. C'est un fichier de *codes* (brr! ça fait froid dans le dos !) mélangeant du html, du css et le langage propre à spip.

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<HTML_>
<HEAD>
<TITLE>
<META>
<META>
<META>
</HEAD>
<BODY>
<DIV id="conteneurGlobal">
<!-->
</BODY>

```

Les textes sont enregistrés dans la base de données et sont "appelés" par les squelettes pour être affichés. La base de données enregistre également l'emplacement des photos et des documents pour que ceux-ci puissent être présentés sur les pages.

Pour accélérer l'affichage des pages, spip utilise un système de cache : le moteur de spip n'accède pas systématiquement à la base de données quand une page est demandée. Il lit parfois les données à partir du "cache".

Il est ainsi parfois nécessaire de demander au moteur un "re-calcul" de la page pour afficher les dernières informations saisies dans la base de données.

Personnalisation des squelettes

Pour personnaliser un squelette, il est nécessaire de connaître la syntaxe des commandes spip et

d'avoir un bon éditeur de texte (par exemple [notepad++](#) sous winxxx ou gedit sous gnome ou nano en ligne de commande)

- la balise <BOUCLE> indique à spip que l'on souhaite interroger la base de données.

exemple 1:

```
<BOUCLE_test(ARTICLES){id_article=1}>
  <h1>#TITRE</h1>
  #TEXTE
</BOUCLE>
```

le code précédent recherche dans la table des articles l'article ayant l'identifiant 1, puis affiche son titre et son texte. On remarque que la table "ARTICLES" est mise entre parenthèses, que le critère "id_article=1" est mis entre accolades et que les éléments tirés de la base de données sont mis en majuscules, précédés du symbole # (dièse).

La balise <h1> est une balise propre au html qui indique que c'est un titre de niveau 1.

On peut imbriquer les boucles entre elles et utiliser des éléments du contexte d'une boucle comme critère d'une autre boucle.

exemple 2 :

```
<BOUCLE_rub_racine(RUBRIQUES){racine}{par num titre}>
  [(#REM) On affiche le titre de la rubrique]
  <h1>#TITRE</h1>
  [(#REM)On sélectionne les articles de la rubrique
    grâce au paramètre id_rubrique présent dans le contexte]
  <BOUCLE_arti(ARTICLES){id_rubrique}{par date}{inverse}>
    <h2>#TITRE</h2>
    [<p>(#CHAPO)</p>]
    [(#REM) On affiche le texte en limitant la taille des images à 300px]
    [(#TEXTE|image_reduire{300})]
  </BOUCLE_arti>
</BOUCLE_rub_racine>
```

On voit apparaître ici la syntaxe complète des balises spip :

```
[texte_optionnel-avant(#MA_BALISE|mon_filtre)texte_optionnel_après]
```

Les textes optionnels ne s'affichent que si MA_BALISE est renseigné dans la base. D'autre part, un "filtre" (une moulinette !) s'applique à la balise.

La balise #REM permet de commenter son code de façon à le rendre plus compréhensible pour soi-même et pour les autres.

Un peu de xhtml et css

xhtml

Une page html se compose au minimum :

- d'un **DOCTYPE** : il indique au navigateur le codage (html 4.0, xhtml 1.0, ...) utilisé pour la page.
- d'une balise "html"
- d'une balise "head"="en-tête" avec une balise "title"="titre" de la page.
- d'une balise "body"="le corps du document"

exemple :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" lang="fr">

<head>
  <meta http-equiv="Content-Type"
  content="text/html; charset=iso-8859-1" />
  <title>
    Largeur fixe (750px, header/menu haut/menu gauche flottant/menu
    droite flottant/contenu/footer.
  </title>
</head>

<body>
  <div id="conteneur">

    <div id="header">
      Header (modèle utilisant les positionnement flottants)<a
      href="http://css.alsacreations.com/Modeles-de-mise-en-page-en-CSS">
      (voir tous les mod&egrave;les)</a>
    </div>
    <div id="haut">
      <ul id="menuhaut">
        <li>

          <a href="#">Menu 1</a>
        </li>
        <li>
          <a href="#">Menu 2</a>
        </li>
        <li>
          <a href="#">Menu 3</a>

        </li>
        <li>
          <a href="#">Menu 4</a>
        </li>
      </ul>
    </div>
    <div id="gauche">
      <p>
```

```
    menu gauche
</p>
<p>
    largeur fixe : 150px avec utilisation de la propriété float:left;
</p>
<ul id="menugauche">
    <li>
        <a href="#">Menu 1</a>

    </li>
    <li>
        <a href="#">Menu 2</a>
    </li>
    <li>
        <a href="#">Menu 3</a>
    </li>
    <li>

        <a href="#">Menu 4</a>
    </li>
</ul>
</div>
<div id="droite">
    <p>
        menu droit
    </p>
    <p>

        largeur fixe : 150px avec utilisation de la propriété float:right;
    </p>
    <ul id="menudroit">
    <li>
        <a href="#">Menu 1</a>
    </li>
    <li>

        <a href="#">Menu 2</a>
    </li>
    <li>
        <a href="#">Menu 3</a>
    </li>
    <li>
        <a href="#">Menu 4</a>

    </li>
    </ul>
</div>
<div id="centre">
    <p>Même avec contenu central court, l'utilisation des flottants combinés à l'utilisation de la propriété clear:both; sur le pied de page, permet à ce dernier de "pousser" le conteneur pour intégrer les éléments en
```

float. Le pied de page est correctement positionné à la suite des parties en float les plus longues.</p>

<p>Si le contenu central est plus long que les menus latéraux, il "pousse" également le pied de page vers le bas.</p>

```
</div>
```

```
<div id="pied">
```

```
    pied de page (se place en dessous des éléments flottants grâce à
clear:both;)
```

```
</div>
```

```
</div>
```

```
</body>
```

```
</html>
```

En xhtml, toutes les balises doivent être fermés et sont écrites en minuscules.

CSS

Css signifie "Cascading Style Sheets"="Feuilles de style en cascade".

Les css permettent de mettre en forme un document html.

exemple :

```
body {
font-family:Verdana, Arial, Helvetica, sans-serif;
font-size: 0.8em;
margin: 0;
padding: 0;
}
p {
margin: 0 0 10px 0;
}
#header {
height: 50px;
background-color: #99CCCC;
}
#haut {
height: 30px;
background-color:#CCCCFF;
}
#conteneur {
width:750px;
margin:0 auto;
background-color:#CCCCFF;
}
#centre {
background-color:#9999CC;
margin-left: 150px;
margin-right: 150px;
```

```
}
#gauche {
float:left;
width: 150px;
}
#droite {
float:right;
width: 150px;
}
#pied {
clear:both;
height: 30px;
background-color: #99CC99;
}
#menuhaut {
list-style-type: none;
margin: 0;
padding:0;
}
#menuhaut li {
display: inline;
}
#menuhaut a {
margin: 0 2px;
color: #000000;
text-decoration: underline;
}
#menuhaut a:hover {
text-decoration: none;
}
#menugauche {
list-style-type: none;
margin: 0;
padding:0;
}
#menugauche li {
margin-bottom: 5px;
}
#menugauche a {
margin: 0 2px;
color: #000000;
text-decoration: underline;
}
#menugauche a:hover {
text-decoration: none;
}
#menudroit {
list-style-type: none;
margin: 0;
padding:0;
}
```



```
#menu droit li {  
margin-bottom: 5px;  
}  
#menu droit a {  
margin: 0 2px;  
color: #000000;  
text-decoration: underline;  
}  
#menu droit a:hover {  
text-decoration: none;  
}
```

Le site [alsacrations](#) présentent d'excellents modèles de mise en page.

Liens

- Site officiel avec une documentation en ligne très bien faite : www.spip.net
- Site de ressources (infos, plugins, ...) - www.spip-contrib.net
- Des squelettes pas chers sur spip-art.net
- [livre xhtml et ccs](#) : Collection tête la première chez Oreilly

From:
<https://wiki.pielo.net/> - **Pielo.net - Wiki**

Permanent link:
<https://wiki.pielo.net/formation-spip>

Last update: **2008/09/28 00:00**

