```
<h1 id="introduction-au-html">Introduction au HTML</h1>
<p>Vous vous demandez s&#xFB;rement ce qu&apos;est un template.</p>
<p>Un template est un fichier que vous pouvez r&#xE9;utiliser afin de
pr&#xE9;senter des informations diff&#xE9;rentes sous un seul et m&#xEA;me
format. Par exemple, vous pourriez avoir envie d&apos;utiliser un template
pour &#xE9;crire une lettre : bien que son contenu varie ou qu&apos;elle
puisse &#xEA;tre adress&#xE9;e &#xE0; des personnes diff&#xE9;rentes, sa
forme reste la m&#xEA;me.</p>
<p>Le format d&apos;un template Django est d&#xE9;crit gr&#xE2;ce &#xE0; un
langage qui s&apos;appelle HTML (c&apos;est le m&#xEA;me HTML que celui dont
nous parlions dans le chapitre un, <strong>Comment fonctionne
l&apos;Internet</strong>).</p>
<h2 id="quest-ce-que-le-html-">Qu&apos;est-ce que le HTML ?</h2>
<p>HTML est un code qui est interpr&#xE9;t&#xE9; par votre navigateur
(Chrome, Firefox ou Safari) et qui permet d&apos;afficher une page web
&#xE0; l&apos;utilisateur.</p>
<p>L&apos;abr&#xE9;viation HTML signifie &#xAB; HyperText Markup Language
&#xBB;. <strong>HyperText</strong> signifie que c&apos;est un type de texte
qui supporte les hyperliens entre les pages. <strong>Markup</strong>
signifie que nous avons pris un document et que nous avons balis&#xE9; le
code pour signifier (ici, au navigateur) comment il faut interpr&#xE9;ter la
page. Le code HTML est construit &#xE0; l&apos;aide de
<strong>balises</strong>, chacune commen&#xE7;ant par <code>&lt;</code> et
finissant par <code>&gt;</code>. Ces balises repr&#xE9;sentent des
<strong>&#xE9;l&#xE9;ments</strong> markup.</p>
<h2 id="votre-premier-template-">Votre premier template !</h2>
<p>Cr&#xE9;er un template signifie cr&#xE9;er un fichier template. Et oui,
encore des fichiers ! Vous aviez d&#xE9;j&#xE0; probablement remarqu&#xE9;
que tout tourne autour des fichiers.</p>
<p>Les templates sont sauvegard&#xE9;s dans le dossier
<code>blog/templates/blog</code>. Tout d&apos;abord, cr&#xE9;ons un dossier
appel&#xE9; <code>templates</code> &#xE0; l&apos;int&#xE9;rieur du dossier
de notre blog. Ensuite, cr&#xE9;ez un autre dossier appel&#xE9;
<code>blog</code> &#xE0; l&apos;int&#xE9;rieur de votre dossier templates
:</p>
<pre><code>blog
&#x2514;&#x2500;&#x2500;&#x2500;templates
    &#x2514;&#x2500;&#x2500;&#x2500;blog
</code></pre><p>Vous pourriez vous demander pourquoi nous avons besoin de
deux dossiers portant tous les deux le nom <code>blog</code>. Comme vous le
d&#xE9;couvrirez plus tard, c&apos;est une convention de nommage qui va nous
faciliter la vie quand les choses vont commencer &#xE0; devenir
compliqu&#xE9;es.</p>
<p>Et maintenant, cr&#xE9;ez un fichier <code>post_list.html</code> (laisser
le vide pour le moment) dans le dossier
<code>templates/blog/blog</code>.</p>
<p>Allons regarder &#xE0; quoi ressemble notre site maintenant : <a
href="http://127.0.0.1:8000/" target="_blank">http://127.0.0.1:8000/</a></p>
<blockquote>
<p>Si vous avez encore l&apos;erreur <code>TemplateDoesNotExist</code>,
essayez de red&#xE9;marrer votre serveur. Allez sur votre ligne de commande
```

```
et arr&#xEA;tez votre server en appuyant simultan&#xE9;ment sur Ctrl+C (les
touches Control et C de votre clavier). Vous pouvez le relancer en tapant la
commande <code>python manage.py runserver</code>.</p>
</blockquote>
<p><img src="images/step1.png" alt="Figure 11.1"></p>
<p>Et voil&#xE0;, il n&apos;y a plus d&apos;erreurs ! Bravo :) Cependant,
notre site ne peut rien faire d&apos;autre pour le moment qu&apos;afficher
une page blanche. La faute &#xE0; notre template que nous avons laiss&#xE9;
vide. Allons corriger &#xE7;a.</p>
<p>Ouvrez le nouveau fichier dans l&#x2019;&#xE9;diteur de code et ajoutez
le morceau suivant :</p>
<p></p><p class="code-label">blog/templates/blog/post_list.html</p><p></p>
<pre><code class="lang-html"><span class="hljs-tag">&lt;<span class="hljs-
name">html</span>&gt;</span>
<span class="hljs-tag">&lt;<span class="hljs-name">body</span>&gt;</span>
    <span class="hljs-tag">&lt;<span class="hljs-name">p</span>&gt;</span>Hi
there!<span class="hljs-tag">&lt;/<span class="hljs-
name">p</span>&gt;</span>
    <span class="hljs-tag">&lt;<span class="hljs-name">p</span>&gt;</span>It
works!<span class="hljs-tag">&lt;/<span class="hljs-
name">p</span>&gt;</span>
<span class="hljs-tag">&lt;/<span class="hljs-name">body</span>&gt;</span>
<span class="hljs-tag">&lt;/<span class="hljs-name">html</span>&gt;</span>
</code></pre>
<p>Alors, &#xE0; quoi ressemble notre site web maintenant ? Allons le
d&#xE9;couvrir : <a href="http://127.0.0.1:8000/"
target="_blank">http://127.0.0.1:8000/</a></p>
<p><img src="images/step3.png" alt="Figure 11.2"></p>
<p>&#xC7;a marche ! Bon boulot :)</p>
<ul>
<li>La balise la plus &#xE9;l&#xE9;mentaire, <code>&lt;html&gt;</code>,
figure toujours au d&#xE9;but de n&apos;importe quelle page web tandis que
<code>&lt;/html&gt;</code> est toujours situ&#xE9;e &#xE0; la fin. Comme
vous pouvez le constater, l&apos;int&#xE9;gralit&#xE9; du contenu de notre
page web est situ&#xE9;e entre la balise de d&#xE9;part,
<code>&lt;html&gt;</code>, et la balise fermante,
<code>&lt;/html&gt;</code></li>
<li><code>&lt;p&gt;</code> est la balise pour les &#xE9;l&#xE9;ments de type
paragraphe. <code>&lt;/p&gt;</code> permet de fermer chaque paragraphe.</li>
</ul>
<h2 id="head-et-body">Head et body</h2>
<p>Chaque page HTML est divis&#xE9;e en deux &#xE9;l&#xE9;ments :
<strong>head</strong> (ent&#xEA;te) et <strong>body</strong> (corps.</p>
<ul>
<li><p><strong>head</strong> est un &#xE9;l&#xE9;ment qui contient des
informations sur le document : son contenu ne s&apos;affichera pas &#xE0;
l&apos;&#xE9;cran.</p>
</li>
<li><p><strong>body</strong> est un &#xE9;l&#xE9;ment qui contient tout le
reste. Son contenu s&apos;affichera &#xE0; l&apos;&#xE9;cran et constituera
notre page web.</p>
```

```
</li>
</ul>
<p>Nous utilisons <code>&lt;head&gt;</code> pour transmettre la
configuration de la page au navigateur tandis que <code>&lt;body&gt;</code>
l&apos;informe sur le contenu de la page.</p>
<p>Par exemple, vous pouvez donner un titre &#xE0; votre page web en
utilisant l&apos;&#xE9;l&#xE9;ment titre dans le <code>&lt;head&gt;</code>
:</p>
<p></p><p class="code-label">blog/templates/blog/post_list.html</p><p></p>
<pre><code class="lang-html"><span class="hljs-tag">&lt;<span class="hljs-
name">html</span>&gt;</span>
    <span class="hljs-tag">&lt;<span class="hljs-
name">head</span>&gt;</span>
        <span class="hljs-tag">&lt;<span class="hljs-
name">title</span>&gt;</span>Ola&apos;s blog<span class="hljs-
tag">&lt;/<span class="hljs-name">title</span>&gt;</span>
    <span class="hljs-tag">&lt;/<span class="hljs-
name">head</span>&gt;</span>
    <span class="hljs-tag">&lt;<span class="hljs-
name">body</span>&gt;</span>
        <span class="hljs-tag">&lt;<span class="hljs-
name">p</span>&gt;</span>Hi there!<span class="hljs-tag">&lt;/<span
class="hljs-name">p</span>&gt;</span>
        <span class="hljs-tag">&lt;<span class="hljs-
name">p</span>&gt;</span>It works!<span class="hljs-tag">&lt;/<span
class="hljs-name">p</span>&gt;</span>
    <span class="hljs-tag">&lt;/<span class="hljs-
name">body</span>&gt;</span>
<span class="hljs-tag">&lt;/<span class="hljs-name">html</span>&gt;</span>
</code></pre>
<p>Sauvegardez votre fichier et actualisez la page.</p>
<p><img src="images/step4.png" alt="Figure 11.3"></p>
<p>Vous avez vu comment le navigateur a compris que &#xAB; Le Blog
d&apos;Ola &#xBB; est le titre de votre page ? Il a interpr&#xE9;t&#xE9;
<code>&lt;title&gt;Le blog d&apos;Ola&lt;/title&gt;</code> et a plac&#xE9;
ce texte dans la barre de titre de votre navigateur (c&apos;est ce titre qui
va &#xEA;tre aussi utilis&#xE9; lorsque vous cr&#xE9;ez un marque-page,
etc.).</p>
<p>Vous avez aussi probablement remarqu&#xE9; que chaque balise ouvrante
poss&#xE8;de sa <em>balise fermante</em>, compos&#xE9;e d&apos;un
<code>/</code>, est qu&apos;elles <em>encadrent</em> les diff&#xE9;rents
&#xE9;l&#xE9;ments. Cela signifie que vous ne pouvez pas fermer une balise
si celles imbriqu&#xE9;es &#xE0; l&apos;int&#xE9;rieur de celle-ci
n&apos;ont pas &#xE9;t&#xE9; ferm&#xE9;es.</p>
<p>Pensez &#xE0; lorsque vous mettez des choses &#xE0; l&apos;int&#xE9;rieur
de bo&#xEE;tes. Vous avez une grosse bo&#xEE;te,
<code>&lt;html&gt;&lt;/html&gt;</code>; &#xE0; l&apos;int&#xE9;rieur de
celle-ci, on trouve une plus petite bo&#xEE;te,
<code>&lt;body&gt;&lt;/body&gt;</code>, qui contient elle-m&#xEA;me
d&apos;autres petites bo&#xEE;tes, <code>&lt;p&gt;&lt;/p&gt;</code>.</p>
<p>Essayez de vous rappeler de cet exemple lorsque vous utilisez les balises
```

```
<em>fermantes</em> et que vous avez des &#xE9;l&#xE9;ments
<em>imbriqu&#xE9;s</em>. Si vous ne suivez pas ces r&#xE8;gles, votre
navigateur risque de ne pas &#xEA;tre capable d&apos;interpr&#xE9;ter votre
code correctement et votre page web sera mal affich&#xE9;e.</p>
<h2 id="personnaliser-votre-template">Personnaliser votre template</h2>
<p>Et si nous en profitions pour nous amuser un peu ? Essayons de
personnaliser notre template ! Voici quelques balises que vous pouvez
utiliser :</p>
<ul>
<li><code>&lt;h1&gt;Titre 1&lt;/h1&gt;</code> pour vos titres les plus
importants</li>
<li><code>&lt;h2&gt;Titre 2&lt;/h2&gt;</code> pour les sous-titres</li>
<li><code>&lt;h3&gt;Titre 3&lt;/h3&gt;</code> ... et ainsi de suite
jusqu&apos;&#xE0; <code>&lt;h6&gt;</code></li>
<li><code>&lt;p&gt;Un paragraphe contenant du texte&lt;/p&gt;</code></li>
<li><code>&lt;em&gt;texte&lt;/em&gt;</code> permet de mettre l&apos;accent
sur une partie du texte</li>
<li><code>&lt;strong&gt;texte&lt;/strong&gt;</code> permet de mettre encore
plus l&apos;accent sur une partie de texte</li>
<li><code>&lt;br&gt;</code> permet d&apos;ins&#xE9;rer un saut de ligne
(vous ne pouvez rien mettre &#xE0; l&apos;int&#xE9;rieur d&apos;un
&#xE9;l&#xE9;ment br et il n&apos;y a pas de balise fermante)</li>
<li><code>&lt;a
href=&quot;https://djangogirls.org&quot;&gt;link&lt;/a&gt;</code> permet de
cr&#xE9;er un lien</li>
<li><code>&lt;ul&gt;&lt;li&gt;premier item&lt;/li&gt;&lt;li&gt;second
item&lt;/li&gt;&lt;/ul&gt;</code> permet de cr&#xE9;er des listes, comme
celle que nous sommes en train de faire !</li>
<li><code>&lt;div&gt;&lt;/div&gt;</code> permet de cr&#xE9;er une section au
sein de la page</li>
</ul>
<p>Voici un exemple d&#x2019;un mod&#xE8;le complet, copiez et collez-le
dans <code>blog/templates/blog/post_list.html</code> :</p>
<p></p><p class="code-label">blog/templates/blog/post_list.html</p><p></p>
<pre><code class="lang-html"><span class="hljs-tag">&lt;<span class="hljs-
name">html</span>&gt;</span>
    <span class="hljs-tag">&lt;<span class="hljs-
name">head</span>&gt;</span>
        <span class="hljs-tag">&lt;<span class="hljs-
name">title</span>&gt;</span>Django Girls blog<span class="hljs-
tag">&lt;/<span class="hljs-name">title</span>&gt;</span>
    <span class="hljs-tag">&lt;/<span class="hljs-
name">head</span>&gt;</span>
    <span class="hljs-tag">&lt;<span class="hljs-
name">body</span>&gt;</span>
        <span class="hljs-tag">&lt;<span class="hljs-
name">div</span>&gt;</span>
            <span class="hljs-tag">&lt;<span class="hljs-
name">h1</span>&gt;</span><span class="hljs-tag">&lt;<span class="hljs-
name">a</span> <span class="hljs-attr">href</span>=<span class="hljs-
string">&quot;/&quot;</span>&gt;</span>Django Girls Blog<span class="hljs-
```

```
tag">&lt;/<span class="hljs-name">a</span>&gt;</span><span class="hljs-
tag">&lt;/<span class="hljs-name">h1</span>&gt;</span>
        <span class="hljs-tag">&lt;/<span class="hljs-
name">div</span>&gt;</span>

        <span class="hljs-tag">&lt;<span class="hljs-
name">div</span>&gt;</span>
            <span class="hljs-tag">&lt;<span class="hljs-
name">p</span>&gt;</span>published: 14.06.2014, 12:14<span class="hljs-
tag">&lt;/<span class="hljs-name">p</span>&gt;</span>
            <span class="hljs-tag">&lt;<span class="hljs-
name">h2</span>&gt;</span><span class="hljs-tag">&lt;<span class="hljs-
name">a</span> <span class="hljs-attr">href</span>=<span class="hljs-
string">&quot;&quot;</span>&gt;</span>My first post<span class="hljs-
tag">&lt;/<span class="hljs-name">a</span>&gt;</span><span class="hljs-
tag">&lt;/<span class="hljs-name">h2</span>&gt;</span>
            <span class="hljs-tag">&lt;<span class="hljs-
name">p</span>&gt;</span>Aenean eu leo quam. Pellentesque ornare sem lacinia
quam venenatis vestibulum. Donec id elit non mi porta gravida at eget metus.
Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut
fermentum massa justo sit amet risus.<span class="hljs-tag">&lt;/<span
class="hljs-name">p</span>&gt;</span>
        <span class="hljs-tag">&lt;/<span class="hljs-
name">div</span>&gt;</span>

        <span class="hljs-tag">&lt;<span class="hljs-
name">div</span>&gt;</span>
            <span class="hljs-tag">&lt;<span class="hljs-
name">p</span>&gt;</span>published: 14.06.2014, 12:14<span class="hljs-
tag">&lt;/<span class="hljs-name">p</span>&gt;</span>
            <span class="hljs-tag">&lt;<span class="hljs-
name">h2</span>&gt;</span><span class="hljs-tag">&lt;<span class="hljs-
name">a</span> <span class="hljs-attr">href</span>=<span class="hljs-
string">&quot;&quot;</span>&gt;</span>Mon second post<span class="hljs-
tag">&lt;/<span class="hljs-name">a</span>&gt;</span><span class="hljs-
tag">&lt;/<span class="hljs-name">h2</span>&gt;</span>
            <span class="hljs-tag">&lt;<span class="hljs-
name">p</span>&gt;</span>Aenean eu leo quam. Pellentesque ornare sem lacinia
quam venenatis vestibulum. Donec id elit non mi porta gravida at eget metus.
Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut
f.<span class="hljs-tag">&lt;/<span class="hljs-name">p</span>&gt;</span>
        <span class="hljs-tag">&lt;/<span class="hljs-
name">div</span>&gt;</span>
    <span class="hljs-tag">&lt;/<span class="hljs-
name">body</span>&gt;</span>
<span class="hljs-tag">&lt;/<span class="hljs-name">html</span>&gt;</span>
</code></pre>
<p>Nous avons cr&#xE9;&#xE9; trois sections &#xE0; l&apos;aide de
<code>div</code>.</p>
<ul>
<li>Le premier <code>div</code> contient le titre de notre blogpost -
```

```
c&apos;est &#xE0; la fois un titre et un lien</li>
<li>Les deux autres <code>div</code> contiennent nos posts avec leur date de
publication, un titre de post <code>h2</code> qui est cliquable ainsi que
deux <code>p</code>s (paragraphe) de texte : un pour la date et l&apos;autre
pour notre post.</li>
</ul>
<p>Ce qui nous donne :</p>
<p><img src="images/step6.png" alt="Figure 11.4"></p>
<p>Yaaay ! Pour l&apos;instant, notre template nous permet seulement
d&apos;afficher les <strong>m&#xEA;mes informations</strong> alors que nous
disions pr&#xE9;c&#xE9;demment qu&apos;il doit nous permettre
d&apos;afficher des informations <strong>diff&#xE9;rentes</strong> utilisant
le <strong>m&#xEA;me format</strong>.</p>
<p>Ce qu&apos;on aimerait pouvoir faire maintenant, c&apos;est
d&apos;afficher les posts que nous avons cr&#xE9;&#xE9;s
pr&#xE9;c&#xE9;demment dans l&apos;interface d&apos;administration de
Django. Penchons-nous l&#xE0; dessus.</p>
<h2 id="une-derni&#xE8;re-chose--d&#xE9;ployer-">Une derni&#xE8;re chose :
d&#xE9;ployer !</h2>
<p>Ne serait-il pas g&#xE9;nial de pouvoir voir tous ces changements en
ligne ? Hop, d&#xE9;ployons &#xE0; nouveau !</p>
<h3 id="commiter-et-pusher-votre-code-sur-github">Commiter et pusher votre
code sur GitHub</h3>
<p>Tout d&apos;abord, allons voir quels sont les fichiers qui ont
chang&#xE9; depuis notre dernier d&#xE9;ploiement (lancez ces commandes dans
votre console locale et non celle de PythonAnywhere) :</p>
<p></p><p class="code-label">command-line</p><p></p>
<pre><code>$ git status
</code></pre><p>Assurez-vous d&apos;&#xEA;tre dans le dossier
<code>djangogirls</code>. Voici la commande qui permet de dire &#xE0;
<code>git</code> d&apos;inclure tout les changements qui ont eu lieu dans ce
dossier :</p>
<p></p><p class="code-label">command-line</p><p></p>
<pre><code>$ git add --all .
</code></pre><blockquote class="clearfix alert alert-info"><strong class="fa
fa-2x fa-edit"></strong>
<p> <code>--all</code> signifie que <code>git</code> va aussi voir si vous
avez supprim&#xE9; des fichiers (par d&#xE9;faut, il ne
s&apos;int&#xE9;resse qu&apos;aux nouveaux fichiers ou &#xE0; ceux
modifi&#xE9;s). Essayez de vous rappeler du chapitre 3 : <code>.</code>
permet de d&#xE9;signer le dossier courant.</p>
</blockquote>
<p>Avant que nous puissions uploader nos fichiers, regardons ce que
<code>git</code> &#xE0; l&apos;intention de faire (tous les fichiers que
<code>git</code> va uploader vont appara&#xEE;tre en vert) :</p>
<p></p><p class="code-label">command-line</p><p></p>
<pre><code>$ git status
</code></pre><p>On y est presque : nous devons maintenant lui dire de
sauvegarder ces changements dans son historique. Nous allons y ajouter un
&quot;message de commit&quot; qui nous permettra de d&#xE9;crire ce qui a
&#xE9;t&#xE9; chang&#xE9;. Vous pouvez mettre ce que vous voulez dans un
```

```
message de commit. G&#xE9;n&#xE9;ralement, il est pr&#xE9;f&#xE9;rable de
mettre quelque chose d&apos;utile qui vous permettra de vous souvenir plus
tard de ce que vous avez fait.</p>
<p></p><p class="code-label">command-line</p><p></p>
<pre><code>$ git commit -m &quot;Modification du HTML du site&quot;
</code></pre><blockquote class="clearfix alert alert-info"><strong class="fa
fa-2x fa-edit"></strong>
<p> N&apos;oubliez pas d&apos;utiliser de doubles guillemets autour de votre
message de commit.</p>
</blockquote>
<p>Une fois que nous avons fait cela, nous pouvons mettre en ligne (pusher)
nos modifications sur GitHub :</p>
<p></p><p class="code-label">command-line</p><p></p>
<pre><code>$ git push
</code></pre><h3 id="puller-les-modifications-sur-pythonanywhere-et-
recharger-son-appli-web">Puller les modifications sur PythonAnywhere et
recharger son appli web</h3>
<ul>
<li>Allez sur la page des <a href="https://www.pythonanywhere.com/consoles/"
target="_blank">consoles de PythonAnywhere</a>. Retournez dans votre
<strong>console Bash</strong> ou ouvrez-en une nouvelle puis tapez la
commande suivante :</li>
</ul>
<p></p><p class="code-label">PythonAnywhere command-line</p><p></p>
<pre><code>$ cd ~/&lt;your-pythonanywhere-domain&gt;.pythonanywhere.com
$ git pull
[...]
</code></pre><p>N&#x2019;oubliez pas de remplacer <code>&lt;your-
pythonanywhere-domain&gt;</code> avec votre propre sous-domaine
PythonAnywhere, sans les chevrons. Votre nom de sous-domaine est normalement
votre nom d&apos;utilisateur PythonAnywhere, mais dans certains cas, il peut
&#xEA;tre un peu diff&#xE9;rent (par exemple si votre nom d&apos;utilisateur
contient des lettres capitales). Donc, si cette commande ne fonctionne pas,
utilisez la commande <code>ls</code> (lister fichiers) pour trouver votre
sous-domaine/nom de dossier, puis <code>cd</code> jusque l&#xE0;.</p>
<p>Maintenant, vous voyez votre code en train d&apos;&#xEA;tre
t&#xE9;l&#xE9;charg&#xE9;. Si vous voulez v&#xE9;rifier que le nouveau code
est arriv&#xE9; sur PythonAnywhere, vous pouvez aller sur la page
<strong>&quot;Files&quot;</strong> et y retrouver les fichiers (vous pouvez
naviguer dans les diff&#xE9;rentes pages de PythonAnywhere depuis le
menu).</p>
<ul>
<li>Pour finir, n&apos;oubliez pas de recharger votre application web :
onglet <a href="https://www.pythonanywhere.com/web_app_setup/"
target="_blank">Web</a> puis cliquez sur le bouton
<strong>Reload</strong>.</li>
</ul>
<p>Retournez sur votre site en cliquant sur l&apos;adresse en haut de la
page : normalement, vous devriez voir la derni&#xE8;re version. Si ce
n&apos;est pas le cas, ce n&apos;est pas grave : n&apos;h&#xE9;sitez pas
&#xE0; demander de l&apos;aider &#xE0; votre coach. :)</p>
```