





















# Partie 4 : Aller plus loin avec WordPress

## La structure des fichiers et dossiers WordPress

Name	Size	Type	Modified
 wp-admin	88 items	Folder	mar 10
 wp-content	7 items	Folder	11:32
 wp-includes	186 items	Folder	mar 10
 index.php	418 bytes	Program	sep 25 2013
 license.txt	19,9 kB	Text	jan 6
 readme.html	9,1 kB	Text	mar 10
 wp-activate.php	5,4 kB	Program	sep 23 2017
 wp-blog-header.php	364 bytes	Program	déc 19 2015
 wp-comments-post.php	1,6 kB	Program	aoû 29 2016
 wp-config.php	3,8 kB	Program	mar 20
 wp-config-sample.php	3,6 kB	Program	mar 10
 wp-cron.php	3,7 kB	Program	aoû 20 2017
 wp-links-opml.php	2,4 kB	Program	nov 21 2016
 wp-load.php	3,3 kB	Program	aoû 22 2017
 wp-login.php	36,6 kB	Program	oct 13 2017
 wp-mail.php	8,0 kB	Program	jan 11 2017
 wp-settings.php	16,2 kB	Program	oct 4 2017
 wp-signup.php	30,1 kB	Program	oct 18 2017
 wp-trackback.php	4,6 kB	Program	oct 23 2017
 xmlrpc.php	3,1 kB	Program	aoû 31 2016

Connaître l'organisation des fichiers dans WordPress est important. La structure est toujours la même, pour un petit site ou pour un gros site : le core ou cœur de WordPress ne change pas.

Les fichiers et dossiers à la racine des sites WordPress ont tous la même structure. À la racine, vous trouverez les dossiers wp-admin, wp-content, wp-includes et les fichiers WordPress.

## Fichiers et dossiers à la racine du site WordPress.

Il ne faut pas modifier ou supprimer certains fichiers ou dossiers natifs de WordPress, en l'occurrence ceux qui font partie du core, le site risquerait de ne plus fonctionner. De plus, la mise à jour de WordPress écrase toutes les modifications.

**Le dossier wp-admin** contient l'ensemble des fichiers concernant l'administration du site. Il ne faut pas modifier des fichiers dans ce dossier.

**Le dossier wp-content** contient l'ensemble des fichiers des thèmes, des extensions et des médias. C'est dans ce dossier que s'effectuent les modifications et personnalisations du site.

**Le dossier wp-includes** contient l'ensemble des fichiers du core de WordPress (classes, fonctions, scripts...). Il ne faut pas modifier des fichiers dans ce dossier.

**Le fichier wp-config.php** est un fichier généré lors de l'installation de WordPress, à partir du fichier wp-config-sample.php. Les informations qui ont servi à la configuration du site se trouvent dans ce fichier : le nom de la base de données, le mot de passe, l'URL de la base de données, le préfixe des tables, les clés de sécurité, les langues... Ce fichier est modifié lors du changement d'hébergeur, lors de l'optimisation du site, du débogage, lors d'un changement de langue... La suppression du fichier wp-config.php réaffiche la page de configuration.

**Les autres fichiers .php** servent au core de WordPress et il ne faut absolument pas les modifier ou les supprimer. Cependant, vous pouvez trouver à la racine du site des fichiers qui ont été générés par une extension, certains pourront être modifiés : .htaccess, robots.txt ou sitemap.xml...

### Le dossier wp-content

**Le dossier wp-content** est le dossier servant à la création de thèmes, d'extensions, de fonctions. Il contient les dossiers languages, plugins, themes et le fichier index.php.

D'autres dossiers apparaissent lors d'ajout d'images, comme le **dossier uploads**, et lors de mises à jour, tel le **dossier upgrade**.

D'autres dossiers peuvent encore apparaître selon l'installation de certaines extensions, comme le **dossier cache**.

Dossiers et fichiers du répertoire wp-content, lors de l'installation de WordPress.

**Le dossier languages** contient les fichiers de langue de WordPress. L'installation manuelle s'effectue en téléchargeant par FTP les fichiers de langue dans ce dossier. Les fichiers de langue ont les extensions .mo et .po/.pot.

**Le dossier plugins** contient l'ensemble des extensions installées. De base, le dossier contient deux extensions : Akismet et Hello Dolly. L'installation manuelle s'effectue en téléchargeant par FTP le dossier contenant les fichiers de l'extension dans le dossier plugins.

**Le dossier themes** contient l'ensemble des thèmes installés. De base, il y a trois thèmes : Twenty Fourteen, Twenty Thirteen, Twenty Twelve. L'installation manuelle s'effectue en téléchargeant par FTP le dossier contenant les fichiers et dossiers du thème dans le dossier themes.

**Le dossier uploads** contient l'ensemble des médias téléchargés. Pour les images, WordPress crée trois formats d'image et les classe selon les paramètres entrés dans l'administration. Par défaut WordPress classe les médias dans des dossiers ayant pour nom l'année, avec à l'intérieur des dossiers ayant pour nom les mois en chiffres (01, 02,... 12). Certaines extensions y créent parfois des dossiers pour y stocker divers fichiers.

**Le dossier upgrade** accueille les fichiers compressés des mises à jour de thèmes ou d'extensions. Avant leur décompression, WordPress les télécharge dans ce dossier. Si la mise à jour se passe mal, vous retrouvez le fichier compressé dans ce dossier. Ce dossier apparaîtra lorsque vous effectuerez une mise à jour.

**Le fichier index.php** sert uniquement à la sécurité du dossier wp-content. Ce fichier vide permet d'afficher une page blanche au lieu de l'arborescence du site en cas de dysfonctionnement. Vous le retrouvez dans différents dossiers.

Source : <http://wordpress.laurentdumoulin.com/premier-pas/la-structure-des-fichiers-et-dossiers-wordpress/>

## Création et personnalisation de thèmes

---

Parmi tous les fichiers et dossiers qui constitue un site WordPress, pour la personnalisation, la création de thème ou de plugins et widgets, vous devrez uniquement aller dans le dossier wp-content.

### Création thème enfant

La documentation du codex de WordPress sur la création d'un thème enfant :

[https://codex.wordpress.org/fr:Th%C3%A8mes\\_Enfant](https://codex.wordpress.org/fr:Th%C3%A8mes_Enfant)

<https://wpmarmite.com/creer-theme-enfant/>

Un thème enfant est une sorte de sous-thème qui permet de personnaliser l'apparence de son site.

Il est conseillé de créer votre thème enfant dès le départ de votre travail sur votre site WordPress. Cela vous évitera de perdre parfois certains réglages si vous l'installez sur un site déjà existant.

Créer un thème enfant est INDISPENSABLE si vous désirez personnaliser votre thème même si ce n'est que pour modifier quelques petits détails. Le thème enfant bénéficie de la notion d'héritage du thème parent. En gros vous bénéficiez des mêmes options que votre thème parent avec lequel votre thème enfant est lié, mais dans votre thème enfant ne vont se trouver que les fichiers que vous avez modifiés. Attention si vous souhaitez modifier un fichier qui dans votre thème parent se trouve dans un ou plusieurs sous-dossiers, il est impératif de copier dans votre thème enfant toute l'arborescence menant au fichier que vous souhaitez modifier. Copier et coller uniquement le fichier que l'on souhaite modifier à la racine du thème enfant s'il est placé dans plusieurs sous-dossiers dans l'arborescence du thème parent ne fonctionnera pas. Il faut respecter le chemin d'accès et l'arborescence du thème parent lorsque l'on copie un fichier que l'on désire modifier dans son thème enfant.

Pourquoi ne pas modifier directement les fichiers dans le thème parent ? Tout simplement parce que les thèmes sont mis à jour régulièrement et il est nécessaire d'effectuer ces mises à jour dès qu'elles sont disponibles pour éviter des failles de sécurité, corriger d'éventuels bugs et assurer la compatibilité avec la dernière version de WordPress.

Si vous faites vos modifications par exemple directement dans les fichiers de votre thème, la mise à jour de celui-ci viendra écraser toutes vos modifications et tout sera à refaire. Grâce au thème enfant vous pouvez mettre le thème à jour en toute sécurité, et vos fichiers modifiés resteront protégés dans votre thème enfant, mais toujours liés à la nouvelle version du thème parent.

### Comment créer un thème enfant

- Positionner vous dans le dossier de votre site dans « wp-content » --> « themes » Vous retrouverez à cet endroit un dossier pour chaque thème installé sur votre site (installé ne veut pas dire activé). Actuellement vous aurez les thèmes « twentyfifteen », « twentyseventeen » et « twentysixteen » qui sont installés par défaut avec la version 4.9.4 de WordPress, ainsi qu'un fichier index.php.

### Remarque importante

Si vous ouvrez le fichier index.php, vous ne trouverez pas de balise fermante ? > pour le PHP. Il ne s'agit pas d'une erreur. Si on fermait cette balise, automatiquement les autres parties de code liées à celle-ci ne s'exécuteraient plus. Donc, **ne fermer pas la balise PHP ? > c'est normal qu'elle ne soit pas présente.**

À cet endroit nous allons créer un nouveau dossier qui va porter le nom du thème parent suivi de la mention "child". Donc par exemple si c'est le thème parent twentyfifteen qui est le plus proche du résultat que l'on veut obtenir et que l'on veut customiser un peu, on va créer un dossier "twentyseventeen\_child" (Attention éviter les espaces, et les caractères accentués et majuscules dans le nom de votre thème enfant). Vous pouvez lui donner un autre nom comme "mon\_theme\_a\_moi" mais l'avantage de reprendre le nom du thème parent suivi de child permet en un coup d'œil de savoir quel est le thème de base.

Pour les utilisateurs de Linux: si vous avez un souci pour créer un dossier, c'est que vous n'avez pas les droits administrateur sur votre dossier. On va arranger cela.

1. Aller dans le dossier de votre serveur où se trouve le dossier de votre site en local.
2. Click droit "Open in terminal"
3. Vous devriez avoir un chemin dans la console qui ressembler à ça

```
user@nb25:~/Desktop/html$
```

ou

```
user@nb25:/opt/lampp/htdocs$
```

4. Taper la commande suivante

```
sudo chmod -R 775 wp_workshop
```

Si la commande précédente ne fonctionne pas et qu'en faisant un clic droit dans le dossier de votre site WordPress dans le dossier de votre serveur local, vous n'avez pas accès au menu contextuel qui vous permet de créer un nouveau dossier ou un nouveau fichier, essayer alors cette commande :

```
sudo chmod -R 777 wp_workshop
```

Ces 2 commandes vous donnent les droits administrateur sur votre dossier WordPress.

5. Vous pouvez à présent créer un dossier "twentyseventeen\_child" dans wp-content --> themes

### **Remarque importante:**

Nous allons appliquer sur notre site le thème enfant, mais le thème parent de ce site devra toujours rester installé sur votre site. Vous ne devez jamais le supprimer car c'est

à l'intérieur de celui-ci que votre thème enfant va chercher la majorité de ses fonctionnalités. Votre thème enfant ne contient que les fichiers que vous avez personnalisés.

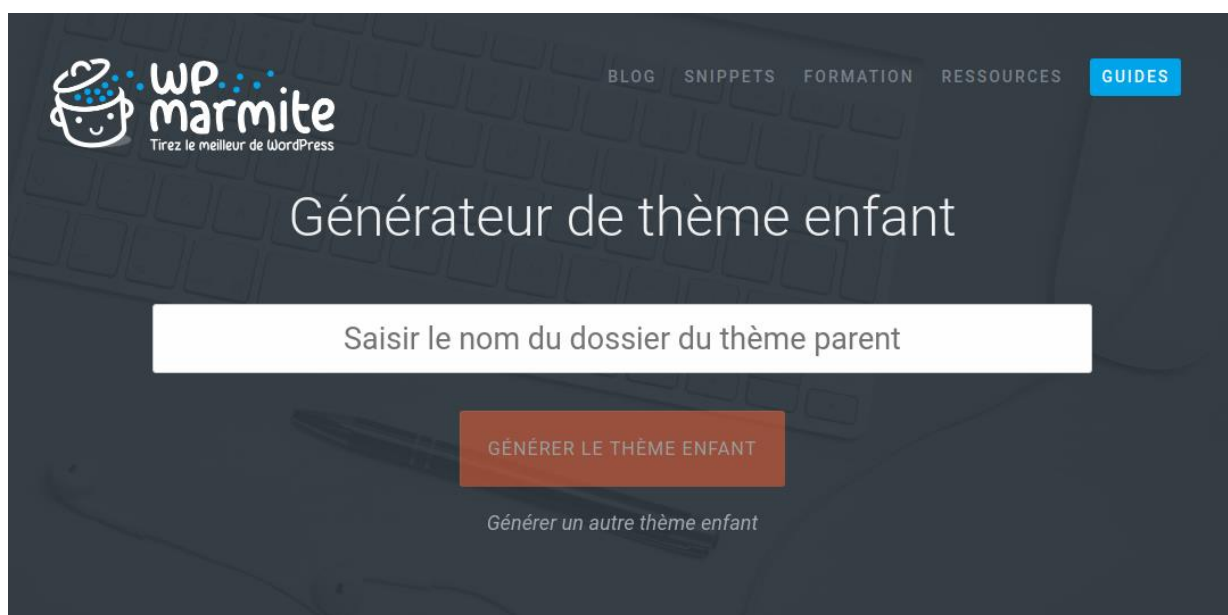
Dans le dossier "twentyseventeen\_child", nous allons placer un fichier **style.css** (obligatoire) qui va permet d'effectuer le lien entre le thème parent et le thème enfant. Ce fichier doit obligatoirement contenir toute une série d'information. Le dossier du thème enfant devra aussi contenir un fichier **screenshot.jpg** (optionnel) (vignette de 1200 x 900 px) qui s'affichera dans votre choix de thème dans l'interface de WordPress. Il s'agit généralement d'une capture d'écran du thème modifié donnant un aperçu visuel du thème après modification. Et enfin un fichier **functions.php** (obligatoire)

### Créer un thème enfant avec le générateur de thème enfant de WP Marmite

Il existe un outil qui vous génère automatiquement ce fichier à cet endroit <https://wpmarmite.com/creer-theme-enfant/>, mais vous verrez que WP marmite qui l'a créé fait clairement sa pub à l'intérieur (c'est le prix à payer pour utiliser cet outil mis à votre disposition gratuitement).

The image shows the interface of the 'Générateur de thème enfant' (Child Theme Generator) tool by WP Marmite. The background is dark with a faint keyboard pattern. At the top left is the WP Marmite logo with the tagline 'Tirez le meilleur de WordPress'. To the right are navigation links: 'BLOG', 'SNIPPETS', 'FORMATION', 'RESSOURCES', and a blue 'GUIDES' button. The main heading is 'Générateur de thème enfant'. Below it is a prompt: 'Entrez l'adresse du site WordPress pour lequel il faut créer un thème enfant'. A large white input field contains the placeholder text 'Saisir l'URL d'un site WordPress'. Below the input field is an orange button labeled 'GÉNÉRER LE THÈME ENFANT'. At the bottom, there is a link that says 'Votre site n'est pas encore en ligne ?'.

Vous pouvez également avoir accès à la création d'un thème enfant même si votre site n'est pas encore en ligne (et ne possède donc pas encore d'URL) en cliquant sur « Votre site n'est pas encore en ligne ? Vous serez redirigé vers une page vous demandant d'indiquer de saisir uniquement le nom du thème parent à la place de l'URL (attention faite un copier-coller du nom du thème parent se trouvant dans le dossier wp-content de votre site WordPress).



Les fichiers générés sont bien faits, mais le nom de WP-Marmite est très présent dans le fichier style.CSS et le logo WP marmite figure sur le screenshot de la vignette du thème enfant.

Vous pouvez bien entendu modifier cela voici pour exemple le code de mes fichiers une fois modifié

### Fichier style.CSS

```
/*
Theme Name: twentyseventeen_child
Description: Theme enfant du thème twentyseventeen
Author: Marie-Ange Bouchat
Author URI: https://github.com/bouchat-marieange
Template: twentyseventeen
Version: 1.0
*/
```

### Fichier screenshot.jpg

screenshot.jpeg (une image format JPG d'une dimension de 1200 x 900 pixels). Vous pouvez dans un premier temps garder celle générée par WordPress ou en créer une vous-même. Lorsque vous aurez personnalisé votre thème, je vous conseille alors de faire une capture d'écran du thème modifié de la recadrer au format 1200 x 900 pixels et de l'utiliser comme vignette pour votre thème enfant (vous aurez ainsi un visuel plus réaliste)

### Fichier functions.php

Ce fichier est parfait tel qu'il est et ne nécessite aucune modification

```
<?php
```

```

/* Chargement de la feuille du style du thème parent */
add_action( 'wp_enqueue_scripts', 'wpchild_enqueue_styles' );
function wpchild_enqueue_styles(){
    wp_enqueue_style( 'wpm-/home/user/Desktop/html/wp_workshop/wp-content/themes-
style', get_template_directory_uri() . '/style.css' );
}

```

Voilà vous avez créé votre premier thème enfant qui contient 3 fichiers: **functions.php** (qui établit le lien avec le thème parent), **screenshot.jpg** (qui sera la vignette affichée pour symboliser votre thème enfant dans l'administration de WordPress), **style.css** (le fichier qui vous permettra de customiser les différentes parties de votre site grâce au CSS au HTML et au PHP').

## Créer un thème enfant manuellement

Il n'est pas nécessaire de se servir d'un générateur de thème enfant, celui-ci vise juste à vous simplifier la vie et à veiller à ce que vous n'oubliez aucune information. Mais vous pouvez créer très simplement un thème enfant manuellement, une fois que vous aurez compris à quoi sert chacun des fichiers et le code qui le compose.

Créer un nouveau dossier dans le dossier thèmes situés dans le dossier wp-content de votre site WordPress. Donner à ce dossier le nom du thème parent suivi de la mention \_child. Exemple : twentyseventeen\_child. Ensuite, créer les fichiers de votre thème enfant et enfin activer votre thème enfant dans l'interface de WordPress.

Un thème enfant doit comme vous le savez maintenant contenir obligatoirement 3 fichiers :

- style.css
- screenshot.jpg
- functions.php

A ceux-ci viendront s'ajouter les fichiers et dossiers des éléments qui seront par rapport au thème parent. On activera notre site enfant à notre thème WordPress. Celui-ci héritera de toutes les fonctionnalités de son thème parent, mais les fichiers qui seront modifiés et placés dans ce dossier auront la priorité sur ceux du thème parent. Vos modifications resteront donc toujours visibles et prioritaires et ne seront jamais écrasées par la mise à jour du thème parent.

### style.css (obligatoire)

```

/*
Theme Name: twentyseventeen_child
Description: Theme enfant du theme twentyseventeen
Author: Marie-Ange Bouchat

```



```
Author URI: https://github.com/bouchat-marieange
Template: twentyseventeen
Version: 1.0
*/
```

Ce fichier est vital. C'est lui qui fait la liaison entre le thème enfant et le thème parent et indique à WordPress, d'aller chercher toutes les fonctionnalités du thème parent sauf pour les fonctionnalités ou fichiers modifiés situés dans le thème enfant qui viendront alors se substituer à celle du thème parent.

A la suite de cette partie mise sous forme de commentaire, mais qui est cependant prise en compte par WordPress, vous pourrez placer toutes les modifications de style que vous souhaitez soit en reprenant les id et class du thème de départ, soit en ajoutant de nouvelles classe et id.

Exemple :

```
/*
Theme Name: twentyseventeen_child
Description: Theme enfant du theme twentyseventeen
Author: Marie-Ange Bouchat
Author URI: https://github.com/bouchat-marieange
Template: twenty seventeen
Version: 1.0
*/

.copyright
{
/* background-color: red; */
text-align: center;
}
```

## functions.php (obligatoire)

Le rôle du fichier functions.php est de mettre en file d'attente les feuilles de style parents et enfant. Configurer le 'style-parent' comme une dépendance veillera à ce que les chargements des feuilles de style du thème enfant se fasse après et donc aie la priorité sur les feuilles de style parent.

```
add_action( 'wp_enqueue_scripts', 'theme_enqueue_styles' );
function theme_enqueue_styles() {
    wp_enqueue_style( 'parent-style', get_template_directory_uri() . '/style.css' );
}
```

Attention, le code du fichier functions.php ci-dessus ne fonctionnera que si votre thème parent utilise un seul style.css principal pour contenir tout le CSS. Si votre

thème a plus d'un fichier .css (par exemple ie.css, style.css, main.css) alors vous devrez vous assurer de maintenir toutes les dépendances du thème parent.

```
<?php

/* Chargement de la feuille du style du theme parent */
add_action( 'wp_enqueue_scripts', 'wpchild_enqueue_styles' );
function wpchild_enqueue_styles(){
    wp_enqueue_style( 'wpm-/home/user/Desktop/html/wp_workshop/wp-content/themes-
style', get_template_directory_uri() . '/style.css' );
}
```

### screenshot.jpg

Le fichier screenshot.jpg est optionnel. Il s'agit d'une image de 1200 px x 900px ayant pour but de vous fournir un visuel du thème enfant que vous allez appliquer à votre site. Vous pouvez soit créer une image en tapant en grand juste un texte reprenant le nom du thème parent suivi de la mention child, ou mieux, une fois les modifications effectuées dans votre thème enfant, faites une capture d'écran et utiliser là comme vignette. Cela donnera immédiatement un aperçu des changements visuels opérés par vos modifications dans le thème enfant.

### IMPORTANT :

Pour que cela fonctionne il faut maintenant aller dans votre administration WordPress "Apparence" --> "Thèmes" et **"Activer" votre thème qui apparaît dans la liste des thèmes installés lorsque vous rafraîchissez la page**. Faites une prévisualisation en direct, si ça marche "Activer le thème". À partir de maintenant votre site fonctionne en se basant sur le thème parent, mais les modifications que vous ferez dans votre thème enfant auront la priorité sur le thème parent et viendront donc remplacer ces parties modifiées par votre version contenue dans le thème enfant.

Voici un petit article bien clair sur le sujet: <https://wpformation.com/theme-enfant-wordpress/>

## Application pratique du thème enfant - changez l'inscription du footer du site.

Maintenant que vous avez un thème enfant, vous allez copier-coller à l'intérieur les parties de site que vous désirez modifier.

Dans ce petit exercice, je vous propose de modifier la phrase par défaut "Fièrement propulsé par WordPress" par un petit copyright personnalisé se mettant automatiquement à jour lorsque l'on change d'année.

Exemple: Athénée Robert Catteau 2018 - © HelloWorld-Web tous droits réservés

1. Rendez-vous dans le dossier de votre thème parent (twentyseventeen)
2. ouvrez le fichier appelé footer.php (qui correspond contient les infos d'affichage de votre bas de page)

A l'intérieur de celui-ci on ne trouve nul par la mention ""Fièrement propulsé par WordPress", celle-ci doit donc se trouver à un autre endroit. En regardant dans le code, on trouve cette ligne qui nous indique où se trouve la mention que l'on veut modifier:

```
get_template_part( 'template-parts/footer/site', 'info' );
```

Cela nous indique que pour changer la mention il va falloir aller dans le dossier template-part/footer/site-info.phpmyadmin

En effet en ouvrant ce fichier site-info.php on trouve le code suivant:

```
<?php
/**
 * Displays footer site info
 *
 * @package WordPress
 * @subpackage Twenty_Seven
 * @since 1.0
 * @version 1.0
 */

?>
<div class="site-info">
    <a href="<?php echo esc_url( __( 'https://wordpress.org/',
'twentyseventeen' ) ); ?>"><?php printf( __( 'Proudly powered by %s',
'twentyseventeen' ), 'WordPress' ); ?></a>
</div><!-- .site-info -->
```

On ne peut pas copier uniquement ce fichier dans notre dossier thème enfant, il faut respecter l'arborescence à partir de la racine du dossier du thème parent (twentyseventeen). Il faut donc copier et coller tout le dossier "template-parts" dans notre dossier thème enfant (twentyseventeen\_child)

4. Ouvrez le fichier avec votre éditeur de texte (Atom, sublime text, ...) site-info.php qui a été copié dans votre thème enfant avec toute son arborescence de dossiers

Nous allons pouvoir maintenant le modifier et enregistrer les modifications

## le code original

```
<?php
/**
 * Displays footer site info
 *
 * @package WordPress
 * @subpackage Twenty_Seventeen
 * @since 1.0
 * @version 1.0
 */

?>

<?php
/**
 * Displays footer site info
 *
 * @package WordPress
 * @subpackage Twenty_Seventeen
 * @since 1.0
 * @version 1.0
 */

?>
<div class="site-info">
    <a href="<?php echo esc_url( __( 'https://wordpress.org/',
'twentyseventeen' ) ); ?>"><?php printf( __( 'Proudly powered by %s',
'twentyseventeen' ), 'WordPress' ); ?></a>
</div><!-- .site-info -->
```

## le code modifié et commenté

```
<?php
/**
 * Displays footer site info
 *
 * @package WordPress
 * @subpackage Twenty_Seventeen
 * @since 1.0
 * @version 1.0
 */

?>
<!-- Cette ligne ajoute dans votre footer un copyright qui va changer
automatiquement chaque année grâce à la fonction PHP date.
On a ajouté en plus de l'id une nouvelle class copyright pour pouvoir la récupérer
dans notre fichier style.css et styliser cet élément-->
<p class="copyright" id="footer-info"> 2015-<?php echo date('Y'); ?> BECODE |
Mentions légales | Site Map</p>
```

```

<!-- On modifie également le "Proudly powered by WordPress" en le remplaçant par
BeCode
et on change le lien qui redirigeait vers le site de WordPress pour rediriger vers
le site de BeCode.org -->
<div class="site-info">
    <a href="<?php echo esc_url( __( 'https://www.becode.org/',
'twentyseventeen' ) ); ?>">

<?php printf( __( 'Proudly powered by %s', 'twentyseventeen' ), 'BeCode' );
?>

</a>
</div><!-- .site-info -->

```

### la modification du fichier style.css du theme enfant pour styliser tout ça

```

/*
Theme Name: twentyseventeen_child
Description: Theme enfant du theme twentyseventeen
Author: Marie-Ange Bouchat
Author URI: https://github.com/bouchat-marieange
Template: twentyseventeen
Version: 1.0
*/

.copyright
{
/* background-color: red; */
text-align: center;
}

.site-info
{
/* background-color: blue; */
width: 100%;
text-align : center;
}

```

Enregistrez vos fichiers et allez dans votre navigateur voir le résultat après avoir rafraîchi la page de votre site et éventuellement vider le cache de votre navigateur. (une extension pratique pour le navigateur chrome pour vider en un click le cache de votre navigateur : <https://chrome.google.com/webstore/detail/clear-cache/cppjkneekbjaeellbfkmgnhonkkjfpdn?hl=fr>)

Bravo ! Vous venez de réaliser votre première personnalisation de thème !!!

## Création d'un nouveau thème

Un thème à la base est composé de nombreux fichiers pour pouvoir prendre en charge toutes les fonctionnalités proposées par WordPress.

Il ne faut pas penser qu'il s'agit juste d'un ou deux fichiers proposant une mise en page de base. Ça c'est de la personnalisation d'un thème existant à l'aide d'un thème enfant.

Je vous conseille pour commencer d'analyser en détail le code présent dans les fichiers des thèmes fournis par défaut par WordPress. Voyer quel est leurs structures, comment les différents fichiers interagissent entre eux, comment sont implémenté les fonctionnalités principales et les différentes zones de l'affichage (header, footer, contenu, zone de menu, zone de widgets...).

Cette première approche vous permettra de déjà avoir une vue d'ensemble sur le fonctionnement d'un thème et des prérequis minimum pour en concevoir un vous-même.
























Une fois cette étape réalisée, il vous faudra maquetter non pas une page de base, mais une série de pages pour votre thème pour pouvoir répondre à l'affichage de différents contenus, article seul, liste articles, pages 404, etc.

Voici un tutoriel qui vous explique comment passer de la maquette HD à la réalisation de votre thème: [https://www.youtube.com/watch?v=dIY73\\_SeGnU](https://www.youtube.com/watch?v=dIY73_SeGnU)

Des outils comme underscores.me peuvent également vous aider à créer automatiquement toute la structure de votre thème avant la phase de personnalisation.

### **Création d'un nouveau thème vierge avec underscores.me**

Utiliser le site <http://underscores.me/> pour créer un dossier zip portant le nom de votre nouveau thème que vous avez entré sur le site. Ce dossier zip contient tous les dossiers et fichiers de base d'un dossier thème vierge. Cela vous évite de devoir créer tous ces fichiers manuellement. Il ne vous reste plus qu'à décompresser l'archive zip générée par le site et placer le dossier décompressé dans la structure de votre site WordPress dans le dossier wp-content/themes.

Name	▲	Size	Type	Modified
 inc		5 items	Folder	12:15
 js		3 items	Folder	12:15
 languages		2 items	Folder	12:15
 layouts		2 items	Folder	12:15
 template-parts		4 items	Folder	12:15
 404.php		1,7 kB	Program	11:15
 archive.php		1,1 kB	Program	11:15
 comments.php		1,9 kB	Program	11:15
 footer.php		955 bytes	Program	11:15
 functions.php		5,0 kB	Program	11:15
 header.php		1,9 kB	Program	11:15
 index.php		1,3 kB	Program	11:15
 LICENSE		18,0 kB	Text	11:15
 page.php		877 bytes	Program	11:15
 phpcs.xml.dist		4,9 kB	Markup	11:15
 readme.txt		1,1 kB	Text	11:15
 README.md		3,1 kB	Text	11:15
 rtl.css		374 bytes	Text	11:15
 screenshot.png		264 bytes	Image	11:15
 search.php		1,2 kB	Program	11:15
 sidebar.php		354 bytes	Program	11:15
 single.php		726 bytes	Program	11:15
 style.css		18,3 kB	Text	11:15

## Création d'extensions (Plugins)

Les plugins ajoutent des fonctionnalités à votre site WordPress. Il y a donc autant de plugins possibles que de fonctionnalités que vous pourriez vouloir intégrer.

Afin de mieux cerner les étapes nécessaires à la création de plugin, je vous renvoie vers ce lien:

<https://www.youtube.com/watch?v=f4dW5xmeuFE>

## Création de widgets

Les widgets sont des composants de navigation qui permettent de faciliter la vie des utilisateurs de votre site. (exemple: articles récents, Archives, Categories,...)

Ces widgets permettent d'avoir un accès direct aux articles récents, aux différentes catégories, etc. c'est en cela qu'il facilite et simplifie la navigation sur votre site.

Voici une vidéo explicative qui est sur la version 3 de WordPress, mais les principes restent identiques: <https://www.youtube.com/watch?v=0m63CxVNffY&t=9s>

Voici un tutoriel plus complet en 2 parties:

<https://www.youtube.com/watch?v=YLdrbuE8mq5>

[https://www.youtube.com/watch?v=IZwFW\\_cr5Xg](https://www.youtube.com/watch?v=IZwFW_cr5Xg)

## Snippets pour personnaliser votre site avec quelques copier-coller de fragment de code

Les snippets sont des petits morceaux de code qui permettent de réaliser des actions ciblées pour organiser votre interface d'administration ou la partie frontend de votre site. Ils permettent de gagner du temps et de personnaliser facilement notre interface d'administration selon nos besoins.

<https://wpmarmite.com/snippets/>