

les étapes pour créer un thème

PRÉPARÉ PAR:

Bernard Imbert

Jean-François Leblanc

Adapté par Mathieu Dupont

Pour le cours 420-964-EM

01

créer les gabarits

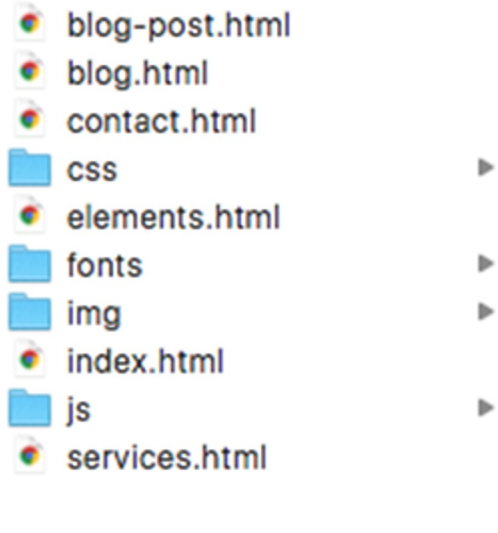
commençons par la base

Avant même de commencer à organiser notre thème WordPress, il faut envisager la structure de notre site web. Pour cette raison, nous allons faire notre HTML et CSS.

- construire les gabarits HTML que nous utiliserons dans notre site
- faire le CSS des gabarits
- faire le JavaScript des gabarits

en bref, faire tout ce que vous avez fait depuis votre cours de Conception Web 1

Quelle serait
la prochaine étape
lorsque nos gabarits
HTML, CSS et JS
sont terminés?



retour sur « the loop »

La boucle sert à afficher le contenu d'une page, de plusieurs articles (*posts*) ou d'un autre type de contenu

```
<!-- s'il y a au moins un article à afficher -->
<?php if ( have_posts() ) : ?>

    <!-- tant qu'il reste un article à afficher, on y avance --> the_post(); -->
    <?php while ( have_posts() ) : the_post(); ?>

        <!-- C'est ici le cœur de la boucle, où on affiche du contenu! -->

    <?php endwhile; ?> <!-- fin du while -->

<?php else : ?> <!-- s'il n'y a aucun article à afficher -->

    <p>Aucun article disponible!</p>

<?php endif; ?> <!-- fin du if -->
```

02

créer les fichiers de base d'un thème

- créer un dossier au nom de votre thème
- créer un fichier `index.php` vide
- créer un fichier `style.css` avec le code nécessaire (mais sans CSS)
- faire une capture pour votre gabarit nommée `screenshot.png`



Il faut maintenant mettre en place notre structure de base.

Prenez le code HTML que vous avez créé (**index.html de votre site statique**) et copiez-le dans **index.php de votre thème**.

intégrer les éléments de base de WordPress

1. remplacer le code statique dans **index.php** par des appels à des **fonctions** PHP de WordPress
(`bloginfo()`, `the_title()`, `the_content()`, etc.)
2. structurer les **inclusions** (header, footer, sidebar)
ne pas oublier l'utilisation de `wp_head()`, `body_class()` et `wp_footer()` aux endroits appropriés
3. utiliser **la boucle**, aussi appelée ***the loop***



1. contenu de base WordPress

Exemple simple d'utilisation de fonctions PHP dans un head :

```
<head>

<meta charset="<?php bloginfo('charset'); ?>">

<title><?php bloginfo('name') ?> <?php wp_title(); ?></title>

<link rel="stylesheet" href="<?php bloginfo('template_url'); ?>/style/style.css"/>

</head>
```

2. structurer les inclusions

Établir les zones répétitives de votre gabarit et en faire des fichiers d'inclusion.

- header
- footer
- sidebar
- publicité
- etc.

Celles-ci sont à titre de référence. Elles peuvent changer d'une interface à l'autre, mais **header** et **footer** seront pratiquement toujours là.

2. structurer les inclusions

header.php

exemple :

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    [...contenu...]
    <?php wp_head(); ?>
  </head>

  <body <?php body_class(); ?>>
    <header>
      [...contenu...]
    </header>
```

2. structurer les inclusions

footer.php

exemple :

```
<footer>
    ©2023, Montréal en lumière
</footer>
<?php wp_footer(); ?>
</body>
</html>
```

2. structurer les inclusions

sidebar.php

exemple :

```
<!-- Zone secondaire de contenu -->
<aside>
    <h3>Les à-côtés</h3>
    [...]
</aside>
```

3. « the loop »

identifier le contenu que nous voulons dynamiser

```
<div class="cards">
  <!-- single item -->
  <div class="card_portfolio">
    <a href="project1.html" class="card_link">
      <div class="card_media">
        
      </div>
      <h2>Nom du projet</h2>
      <p>Lorem ipsum sit dolor amet.</p>
      <a href="#lien">En savoir plus</a>
    </a>
  </div>
  <!-- single item -->
  <div class="card_portfolio">
    <a href="project1.html" class="card_link">
      <div class="card_media">
        
      </div>
      <h2>Nom du projet</h2>
      <p>Lorem ipsum sit dolor amet.</p>
      <a href="#lien">En savoir plus</a>
    </a>
  </div>
  <!-- single item -->
  <div class="card_portfolio">
    <a href="project1.html" class="card_link">
      <div class="card_media">
        
      </div>
      <h2>Nom du projet</h2>
      <p>Lorem ipsum sit dolor amet.</p>
      <a href="#lien">En savoir plus</a>
    </a>
  </div>
</div>
```

3. « the loop » remplacer ce contenu à l'aide de la boucle

```
<div class="cards">
  <?php if (have_posts()): ?>
    <?php while (have_posts()) : the_post(); ?>

      <!-- single item -->
      <div class="card_portfolio">
        <a href="project1.html" class="card_link">
          <div class="card_media">
            <?php the_post_thumbnail('vignette'); ?>
          </div>
          <h2><?php the_title(); ?></h2>
          <p><?php the_excerpt(); ?></p>
          <a href="<?php the_permalink(); ?>">Suite</a>
        </a>
      </div>

    <?php endwhile; ?>
  <?php endif; ?>
</div>
```


03

les fonctions de WordPress

**activons les différentes options
requisies pour notre thème**

les fonctions de WordPress

1. créer le fichier de base
2. activer nos options de base
(images mises en avant, longueur du résumé, etc.)
3. créer nos menus

1. créer le fichier de base

→ le fichier `functions.php`

2. activer les options de bases

- images mises en avant
- taille d'image
- etc.

Assurez-vous d'activer seulement les options qui vous seront utiles. Il ne faut pas polluer votre interface d'administration avec plein de matériel inutile.

```
<?php
```

```
// support image à la une
```

```
add_theme_support('post-thumbnails');
```

```
set_post_thumbnail_size(800, 480, true);
```

```
// ajouter un nouveau format d'image
```

```
add_image_size('vignette', 220, 100, true);
```

```
// etc.
```

3. créer les menus

n'oubliez pas les étapes de création d'un menu

→ créer des zones de menu

`functions.php`

→ afficher ces zones de menu

`gabarits.php`

→ créer les menus et les affecter aux zones

`interface d'administration`

Encore une fois, assurez-vous de ne créer que les zones dont vous aurez réellement besoin.

3. créer les menus

créer les zones de menu

dans functions.php

```
register_nav_menus(array(  
    'menu_principal' => 'Menu principal',  
    'menu_footer' => 'Menu du pied de page',  
    'menu_sociaux' => 'Menu réseaux sociaux',  
));
```

3. créer les menus afficher les zones de menu dans les gabarits

```
<header>
  <nav class="main-menu">
    <ul>
      <li><a href="index.html">Home</a></li>
      <li><a href="projets.html">Portfolio</a></li>
      <li><a href="blog.html">Blog</a></li>
      <li><a href="contact.html">Contact</a></li>
    </ul>
  </nav>
</header>
```



```
<header>
  <?php wp_nav_menu(array(
    'theme_location' => 'menu_principal',
    'container' => 'nav',
  )); ?>
</header>
```

3. créer les menus

créer les menus et les affecter aux zones

Nom du menu

Menu 1

Enregistrer le menu

Structure du menu

Glissez chaque élément pour les placer dans l'ordre que vous préférez. Cliquez sur la flèche à droite de l'élément pour afficher d'autres options de configuration.

Page 1

Page

Page 2

Page

Page 3

Page

Page 4

Page

Catégorie 1

Catégorie

Catégorie 2

Catégorie

Réglages du menu

Ajoutez automatiquement des pages

☐ Ajouter automatiquement les pages de premier niveau à ce menu

Afficher l'emplacement

☒ Menu principal

☐ Menu réseaux sociaux

Supprimer le menu

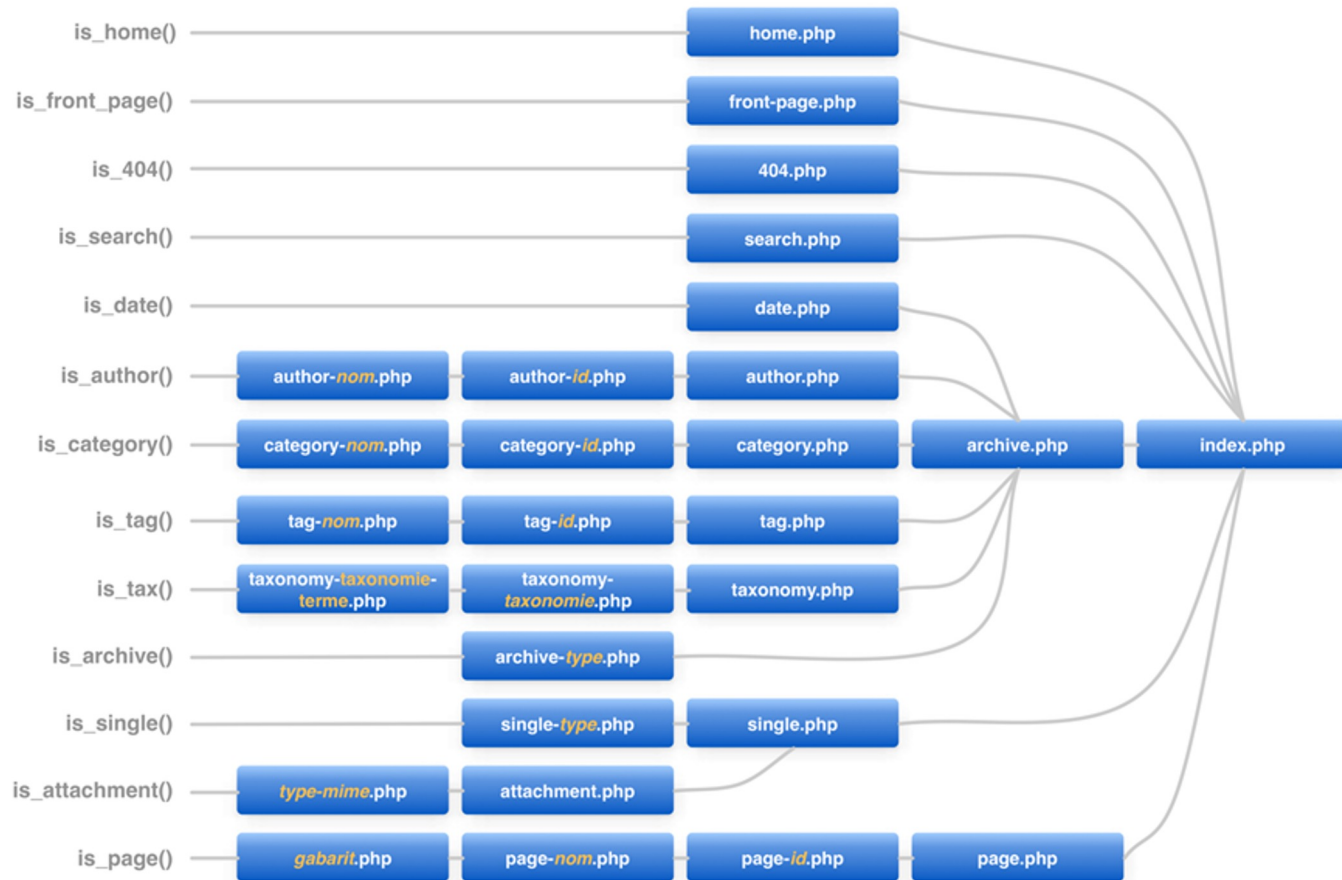
Enregistrer le menu

04

les gabarits

**notre première page est pas mal...
il faut maintenant faire les autres!**

articles



pages

les gabarits

Il faut maintenant faire le bon choix des gabarits.

Tel que vu dans le cours, faites le bon choix en fonction de votre projet.

- créez le fichier **.php** de votre gabarit
- copiez le contenu de votre **index.php** dans ce nouveau gabarit
- utilisez le fichier **.html** correspondant (de votre site statique) pour apporter les modifications à ce nouveau gabarit
- ajoutez ou modifiez des **fonctions** ou des **boucles** afin de remplacer tout votre contenu statique par celui provenant de votre base de données

les gabarits

exemple : **page.php**

Une fois le contenu de votre **index.php** copié dans **page.php** ,
identifiez le contenu à modifier avec le gabarit **.html** de votre
site statique et emmenez-le dans **page.php** afin d'y inclure vos
boucles pour afficher le bon contenu.

On a fait le tour!

Vous avez maintenant créé votre propre thème avec tous les gabarits nécessaires.