

WordPress votre propre thème

(en PHP)

Semaine 10

PRÉPARÉ PAR:

Bernard Imbert

Jean-François Leblanc

Adapté par Mathieu Dupont

Pour le cours 420-964-EM

en PHP

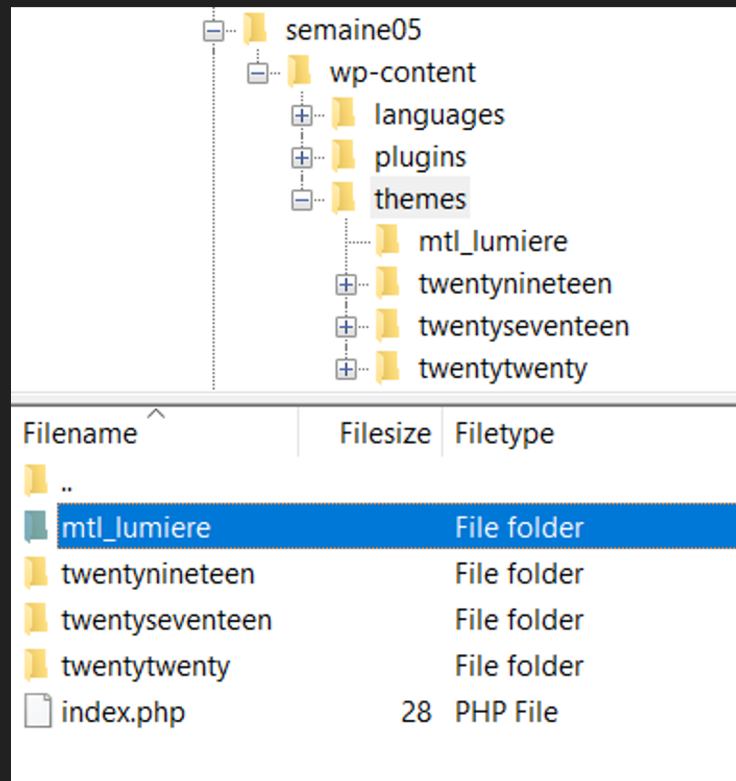
les bases

pour créer un nouveau thème

il suffit de créer un sous-dossier dans **wp-content/themes/** et d'y placer les 2 fichiers magiques :

→ **index.php**

→ **style.css**



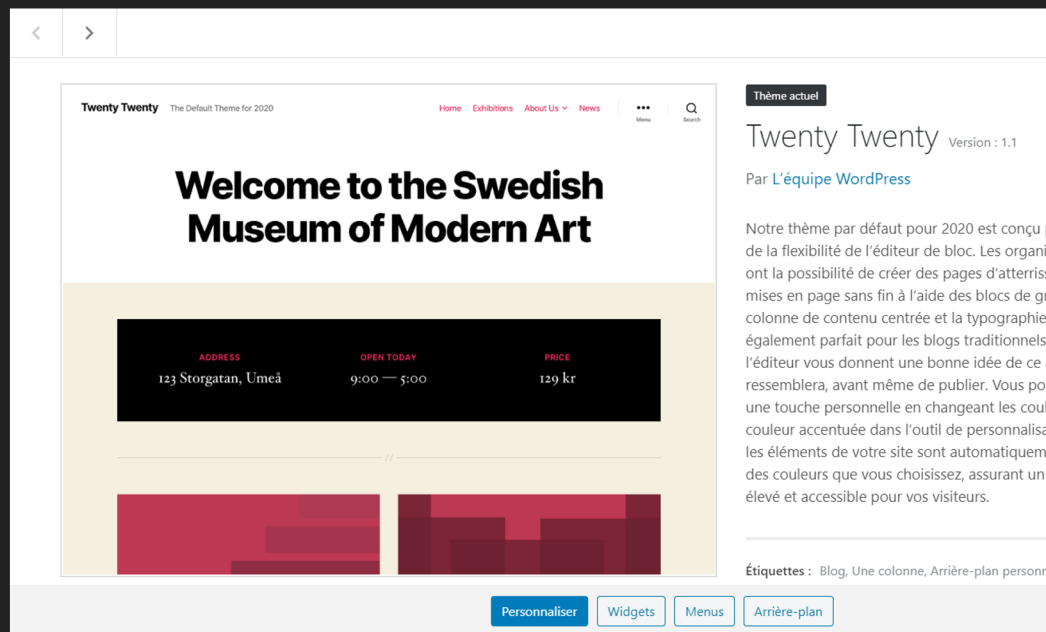
capture d'écran de votre thème

bien que optionnelle, il est
fortement recommandé d'en
produire une :

→ index.php

→ style.css

→ screenshot.png



feuille de style

afin que votre thème s'affiche dans
Apparence > thème, vous devez inscrire les
informations suivantes dans l'en-tête du
fichier **style.css** tout en haut

```
/*  
Theme Name: Twenty Twenty  
Author: the WordPress team  
Author URI: http://wordpress.org/  
Description: The 2020 theme for WordPress ...  
Version: 1.0  
Tags: white, black, minimalist  
Text Domain: twentytwenty  
*/
```



Version

Author

Theme Name

Description

Tags

Twenty Twenty

Version : 1.1

Par L'équipe WordPress

Notre thème par défaut pour 2020 est conçu pour tirer pleinement parti de la flexibilité de l'éditeur de bloc. Les organisations et les entreprises ont la possibilité de créer des pages d'atterrissage dynamiques avec des mises en page sans fin à l'aide des blocs de groupe et de colonne. La colonne de contenu centrée et la typographie affinée le rendent également parfait pour les blogs traditionnels. Les styles complets de l'éditeur vous donnent une bonne idée de ce à quoi votre contenu ressemblera, avant même de publier. Vous pouvez donner à votre site une touche personnelle en changeant les couleurs d'arrière-plan et la couleur accentuée dans l'outil de personnalisation. Les couleurs de tous les éléments de votre site sont automatiquement calculées en fonction des couleurs que vous choisissez, assurant un contraste de couleurs élevé et accessible pour vos visiteurs.

Étiquettes : Blog, Une colonne, Arrière-plan personnalisé, Couleurs personnalisées,

Personnaliser

Widgets

Menus

Arrière-plan

Twenty Twenty

The Default Theme for 2020

Thème actuel

123 Storgatan, Umeå

9:00

Welcome to the Swedish Museum of Modern Art

les fonctions PHP

vous avez créé votre premier thème

Cependant, vous constaterez qu'aucun contenu ne s'affiche. Que devrait contenir le fichier **index.php**?

- le fichier `index.php` contient un mélange de `code PHP` et de `code HTML`. C'est de cette façon que sont construits les gabarits WordPress
- les `fonctions PHP propres à WordPress` seront donc utilisées pour afficher le contenu entré dans le système et le HTML pour la structure de la page
- il faut notamment utiliser ces fonctions pour indiquer correctement où trouver la feuille de style, les images, etc.

la fonction bloginfo()

- `bloginfo()` est une fonction PHP spécifique à WordPress
- C'est ce qu'on appelle dans le jargon WordPress un `marqueur de modèle` (template tags)
- La fonction `bloginfo()` permet d'obtenir du contenu dynamiquement grâce à des `paramètres`

principaux paramètres de bloginfo()

→ `bloginfo('name');`
titre du site

→ `bloginfo('charset');`
jeu de caractère

→ `wp_title();`
titre de la page

→ `bloginfo('stylesheet_url');`
lien vers la feuille de style (style.css)

→ `bloginfo('template_url');`
lien vers le dossier du thème actif

→ `bloginfo('url');`
lien absolu du site



Réglages généraux

Titre du site

Laboratoire semaine 05

Slogan

Un site utilisant WordPress

En quelques mots, décrivez la raison d'être de ce site.

```
<head>
```

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

```
<title>
```

```
<?php bloginfo('name') ?> <?php wp_title() ?>
```

```
</title>
```

```
<link rel="stylesheet" href="<?php bloginfo('template_url') ?>/styles/mes-styles.css">
```

```
</head>
```

façon optimale d'inclure
notre feuille de style

ajouter des images à notre thème?

il faut d'abord faire la distinction entre images dynamiques et statiques :

- **images dynamiques** : images qui peuvent être ajoutées par les éditeurs du site Web via l'interface administrateur
- **images statiques** : images qui sont ajoutées par le développeur du site et qui ne peuvent être modifiées que par lui (non accessibles via le CMS)

ajouter des images statiques

- placer un dossier `images` dans le dossier de votre thème
- récupérez le chemin jusqu'au dossier de votre thème et ajoutez-le à l'attribut `src` de l'image

```

```

```
<!-- Résultats dans le navigateur en vérifiant la source -->
```

```

```

- ceci s'applique seulement aux images statiques. Nous verrons plus tard le fonctionnement pour les images dynamiques

**récupération
du contenu**

comment récupérer et intégrer les contenus de WordPress?

- il faut utiliser une séquence de lignes de code appelée « **la boucle** » ou « **the loop** »
- par défaut, la boucle affiche les 10 derniers articles. Cela peut-être modifié dans la section **Réglages > lecture**

Syntaxe alternative de PHP

avant d'aller plus loin, vous devez
comprendre la syntaxe alternative de PHP

php

```
<?php
    if (true) {
        echo "<p>Test</p>";
    }
?>
```

php + html

```
<?php if (true) : ?>
    <p>Test</p>
<?php endif; ?>

<?php if (true) { ?>
    <p>Test</p>
<?php } ?>
```

« the loop »

```
<!-- s'il y a au moins un article à afficher -->
```

```
<?php if (have_posts()) : ?>
```

```
    <!-- tant qu'il y a un nouvel article à afficher, roule la boucle -->
```

```
    <?php while (have_posts()) : the_post(); ?>
```

```
        <!-- structure html pour afficher le 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 -->
```

plusieurs options possibles dans « la boucle »

→ `the_title();`
titre

→ `the_ID();`
identifiant unique

→ `the_excerpt();`
résumé

→ `the_content();`
contenu

→ `the_post_thumbnail();`
image à la une

`the_category();`
catégorie(s)

→ `the_tags();`
mot(s)-clef(s)

→ `the_date()` **ou** `the_time();`
date de publication

→ `the_permalink();`
permalien



intégrer le contenu de vos articles (post)

```
<?php if(have_posts()) : ?>
    <?php while (have_posts()) : the_post(); ?>

        <article <?php post_class(); ?> id="post-<?php the_ID();?>">
            <h3><?php the_title(); ?></h3>
            <?php the_post_thumbnail('medium'); ?>
            <p class="info">Prix: <?php the_field('prix'); ?></p>
            <p class="info">Lieux: <?php the_field('lieu'); ?> </p>
            <?php the_excerpt(); ?>
            <a href="<?php the_permalink(); ?>" class="more-link">Suite </a>
        </article>

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

les gabarits WordPress

identique à ce qu'on a déjà vu



For oEmbeds: `embed-{post-type}-{post_format}.php` → `embed-{post-type}.php` → `embed.php` → `wp-includes/theme-compat/embed.php`

Primary Template
 Secondary Template
 Variable Template
 Page Type

les fichiers en inclusions

une structure réutilisable et modulaire

header.php

index.php

sidebar.php

footer.php

les fichiers en inclusion

- en-tête : header.php
- pied de page : footer.php
- zone secondaire : sidebar.php
- formulaire de recherche : searchform.php
- commentaires : comments.php
- etc.

appeler les fichiers en inclusion

→ `get_header();`
`header.php`


→ `get_footer();`
`footer.php`

→ `get_sidebar();`
`sidebar.php`

→ `get_search_form();`
`searchform.php`

→ `comments_template();`
`comments.php`

→ `get_template_part('composante');`
`composante.php`



utilisera le fichier
composante.php à la
base de notre thème

le fichier header.php

- ce fichier contiendra le contenu qui débutera chaque page
- cette page doit contenir au minimum les en-têtes html nécessaires jusqu'à l'ouverture du <body>
- remarquez l'ajout des fonctions `wp_head()` et `body_class()`
- `wp_head()` est un hook qui permettra à WordPress de venir incruster du code pour ses propres besoins (JavaScript, CSS, etc.). C'est primordial de le mettre.
- `body_class()` ajoute des classes utiles

header.php

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



bien que les balises `<html>`
et `<body>` soient ouvertes ici,
elles ne seront fermées que
dans `footer.php`

le fichier footer.php

- ce fichier contiendra le contenu qui terminera chaque page
- cette page doit contenir au minimum les balises de fermeture du html, soit `</body>` et `</html>`
- il faut aussi utiliser la fonction `wp_footer()` avant la fermeture de la balise body

footer.php

```
<footer>
```

```
    ©2022, Montréal en lumière - Exercice dans le cours Programmati...
```

```
</footer>
```

```
<?php wp_footer(); ?>
```

```
</body>
```

```
</html>
```

A white rectangular box containing the text "elles sont là!". Two white arrows originate from the top-left corner of the box. One arrow points diagonally upwards and to the left, ending at the closing tag "</body>". The other arrow points diagonally upwards and to the left, ending at the closing tag "</html>".

elles sont là!

le fichier sidebar.php

- ce fichier contiendra le contenu qui se situera dans la colonne latérale
- le sidebar sert généralement à l'affichage de **widgets** (bloc de texte, liste de catégories, formulaire de recherche, un calendrier, etc.), de menus ou de contenu secondaire
- il n'y a aucun code obligatoire qui doit apparaître dans ce fichier

sidebar.php

```
<!-- Zone secondaire de contenu -->
```

```
<aside>
```

```
    <h3>Les à-côtés</h3>
```

```
    [...]
```

```
</aside>
```

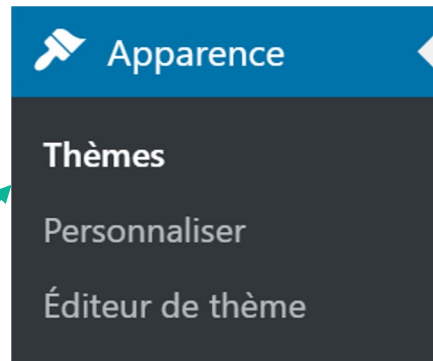
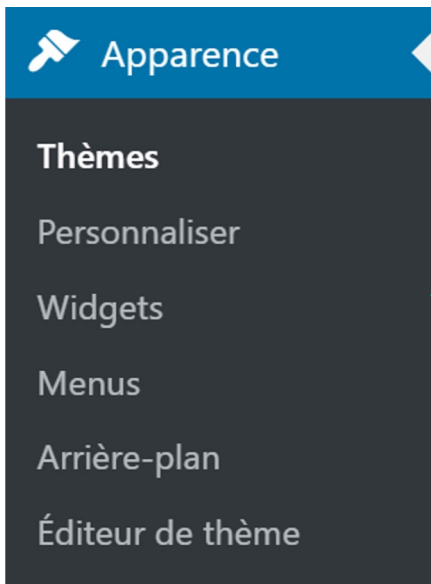
index.php révisé avec les inclusions

```
<?php get_header(); ?>
<div id="banniere">
    (...)
<div id="principal">
    <?php if (have_posts()) : ?>
        <?php while (have_posts()) : the_post(); ?>
            <article>[...]</article>
        <?php endwhile; ?>
    <?php else : ?>
        <p>Oops! Il n'y a rien!</p>
    <?php endif; ?>
</div>
<?php get_sidebar(); ?>
<?php get_footer(); ?>
```

les fonctions WordPress

elles sont rendues où, mes
images à la une?

**certaines options ne sont
affichées que lorsque les
thèmes les supportent**



Pour que les menus et les widgets soient disponibles, le thème doit déclarer qu'il les supporte.

C'est possible en utilisant le fichier **functions.php** dont nous parlerons bientôt.

comprendre le fichier functions.php

s'il est présent à la racine du thème, ce fichier PHP sera automatiquement chargé et exécuté à chaque requête

Il permet :

- de spécifier les fonctionnalités que supportera notre thème (images à la une, widgets, menus, etc.)
- d'ajouter des **filtres** et des **actions** modifiant dynamiquement le rendu des pages

mise en garde

afin d'éviter beaucoup de douleur potentielle, il est fortement recommandé de ne pas mettre la balise de fermeture dans le fichier functions.php

```
<?php
```

```
// insérez le code PHP ici
```

```
// Ne mettez pas la balise de fermeture
```

en fait, on ne devrait jamais mettre la balise de fermeture **?>** dans les fichiers qui ne contiennent que du PHP



fonctionnalités supportées

la fonction `add_theme_support()` permet de spécifier quelles fonctionnalités sont supportées par votre thème

→ pour autoriser l'utilisation des **images en avant**:

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

→ pour que WordPress crée des copies des images à une dimension donnée :

```
set_post_thumbnail_size(800, 480);  
add_image_size('vignette', 220, 180);  
add_image_size('xxl', 2200, 1800);
```

`<extension-regenerate-thumbnails>`

attention, ce n'est pas du vrai code



Regenerate Thumbnails

[Installer](#)

Recrée les miniatures pour une ou plusieurs de vos images téléversées. Utile si vous changez de tailles d'images ou de thème.

[Plus de détails](#)

Par [Alex Mills \(Viper007Bond\)](#)



1 million et + installations
actives

Dernière mise à jour : il y a 2 semaines

✓ Compatible avec votre version de WordPress

Lorsque vous modifiez les grandeurs de vos images. Cette petite extension est très intéressante pour vous assurer que WordPress régénère les images au bon format.

regenerate thumbnails

→ toutes d'un coup

 Outils

Outils disponibles

Importer

Exporter

Santé du site

Exporter les données

Effacer les données

Régénérer les miniatures d'images

Régénérer les miniatures pour toutes les 9 pièces jointes.

Régénérer les miniatures pour les 9 images mises en avant uniquement.

Tailles de miniatures

Voici toutes les tailles de miniatures qui sont actuellement enregistrées :

thumbnail : 150×150 pixels (Recadré pour correspondre.)

medium : 300×300 pixels (Redimensionné proportionnellement pour respecter les tailles)

medium_large : 768×0 pixels (Redimensionné proportionnellement pour respecter les tailles)

large : 1024×1024 pixels (Redimensionné proportionnellement pour respecter les tailles)

1536x1536 : 1536×1536 pixels (Redimensionné proportionnellement pour respecter les tailles)

2048x2048 : 2048×2048 pixels (Redimensionné proportionnellement pour respecter les tailles)

→ une à la fois

☐



Shenzhen

Shenzhen.jpg

[Modifier](#) | [Supprimer définitivement](#) | [Afficher](#) | [Régénérer les miniatures](#)

comme toute bonne chose a une fin

Une fois terminé, vous pouvez désinstaller l'extension **Regenerate Thumbnails**.



Bonnes pratiques

Toujours effacer les extensions et les images qui ne servent plus. Vos sauvegardes en seront plus légères et vous aurez plus d'espace libre sur votre serveur.

`</extension-regenerate-thumbnails>`

on revient aux choses sérieuses

retour à functions.php

il est aussi possible de spécifier le nombre maximum de mots pour l'affichage du texte résumé (`the_excerpt`)

dans functions.php :

```
function new_excerpt_length($length) {  
    return 20; // Nombre maximum de mots  
}  
  
add_filter('excerpt_length', 'new_excerpt_length');
```

ajouter un menu

actions

la fonction `add_action()` permet d'ajouter une opération personnalisée à une certaine étape du processus de génération de la page

```
function ma_fonction() {  
    // insérez le code PHP ici  
}  
  
// la plupart des actions liées au thème  
// et extension pourront utiliser l'action init  
add_action('init', 'ma_fonction');
```

codex: référence `add_action()`

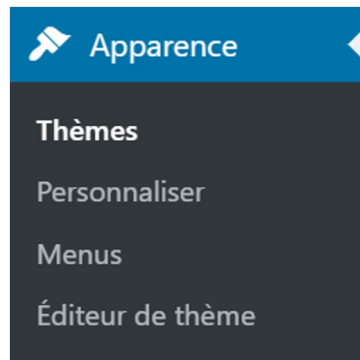
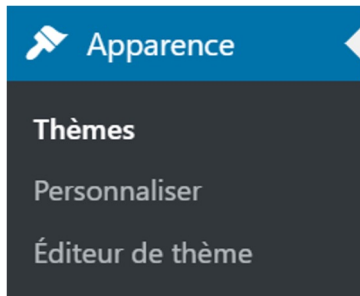


enregistrer des emplacements pour les menus

- lorsque vous utilisez l'interface WordPress pour ajouter un menu ou un widget, vous indiquez au logiciel l'endroit où le menu/widget devrait s'afficher (ex.: sidebar, footer 1, menu_principal, etc.)
- pour pouvoir permettre cela dans votre thème, vous devez enregistrer des emplacements prévus à cet effet et ajouter le code dans votre thème aux endroits désirés afin d'afficher le contenu de ces nouveaux emplacements

enregistrer des emplacements pour les menus

Par défaut, WordPress n'active pas de menu dans un thème personnalisé. La première étape consiste donc à activer les options de menu de WordPress dans `functions.php`



1. l'indiquer à WordPress

→ enregistrer un emplacement de menu

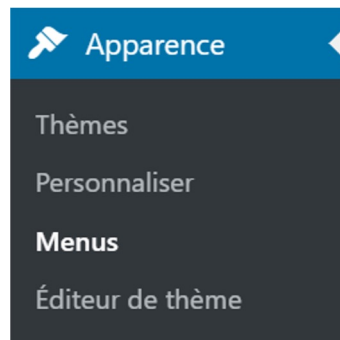
```
function creer_menu() {  
    register_nav_menu('menu_principal', 'Menu principal');  
}  
add_action('init', 'creer_menu');
```

→ enregistrer plusieurs emplacements de menus

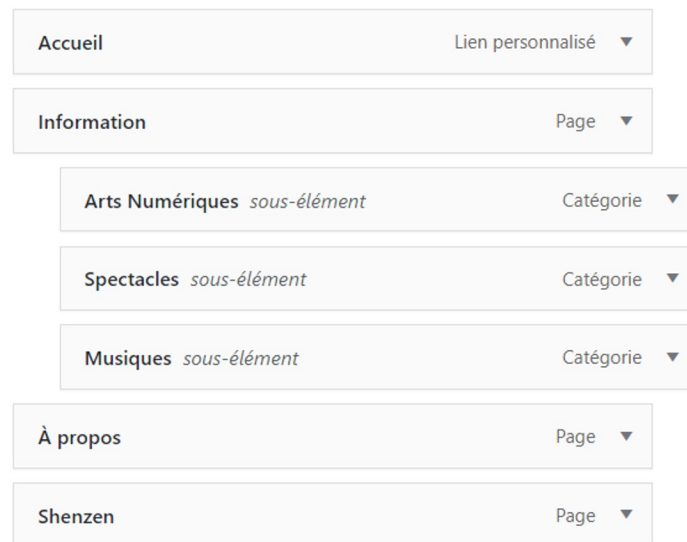
```
function creer_menu() {  
    register_nav_menus(array(  
        'menu_principal' => 'Menu principal',  
        'menu_footer' => 'Menu du pied de page',  
        'menu_sociaux' => 'Menu réseaux sociaux',  
    ));  
}  
add_action('init', 'creer_menu');
```

2. création et assignation du menu

il faut ensuite aller créer notre premier menu dans l'interface d'administration



n'oubliez pas
d'assigner votre menu
à un emplacement



Réglages du menu

Ajoutez automatiquement des pages

☐ Ajouter automatiquement les pages de premier niveau à ce menu

Afficher l'emplacement

- ☒ Menu principal
- ☐ Menu du pied de page
- ☐ Menu réseaux sociaux

3. sélection d'un emplacement

Le menu est maintenant disponible, mais rien ne s'affiche dans notre page.

Ne vous inquiétez pas, c'est normal! Nous devons maintenant lui dire où nous voulons que le menu s'affiche. Nous appelons ça une zone d'emplacement.

Pour déterminer une zone dans votre thème, vous devrez mettre le code suivant à l'endroit désiré dans votre gabarit :

```
<?php wp_nav_menu(array(  
    'theme_location' => 'menu_principal')  
); ?>
```



Houston! on a un menu

la structure par défaut du menu affiché sera :

```
<div class="menu-nom-menu-container">
  <ul id="menu-nom-menu" class="menu">
    <li id="menu-item-id-item" class="menu-item">
      <a href="lien">Texte</a>
    </li>
  </ul>
</div>
```

codex: référence
wp_nav_menu()



voici les différentes classes (CSS) qui sont générées par WordPress

- **.menu-nom-menu-container**: sur le conteneur du menu (<div> ou <nav>)
- **.menu**: sur le contenant le menu
- **.sub-menu**: sur le contenant les pages enfants
- **.menu-item**: sur chaque du menu
- **.current-menu-item**: sur le de la page courante
- **.current-menu-parent**: sur le parent de la page courante (lorsque sous-nav)
- **.current-menu-ancestor**: sur les ancêtres de la page courante (lorsque sous-sous-nav)
- **.menu-item-home**: sur les de la page d'accueil



Lors de l'intégration html/css, prenez tout-de-suite en considération le nom des classes générées par WordPress.

Vous sauverez énormément de temps.

De toute façon, c'est une belle nomenclature de classes de menu, pourquoi ne pas toujours l'utiliser?

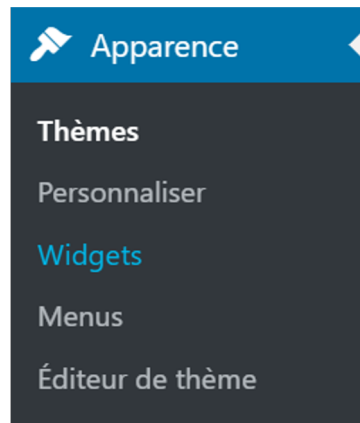
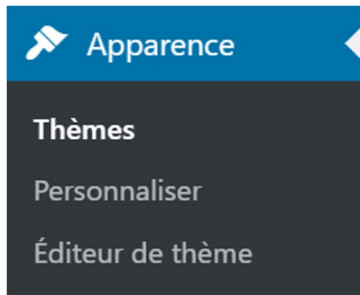
ajouter un widget
sinon, à quoi sert ma barre latérale?

enregistrer des emplacements pour les widgets

- lorsque vous utilisez l'interface WordPress pour ajouter un **menu** ou un **widget**, vous indiquez au logiciel l'endroit où le menu/widget devrait s'afficher (ex.: sidebar, footer 1, menu_principal, etc.)
- pour pouvoir permettre cela dans votre thème, vous devez **enregistrer des emplacements** prévus à cet effet et ajouter le code dans votre thème aux endroits désirés afin d'afficher le contenu de ces nouveaux emplacements
- **en gros, c'est pareil que pour les menus**

enregistrer des emplacements pour les widgets

Par défaut, WordPress n'active pas de widget dans un thème personnalisé. La première étape consiste donc à activer les options de widget de WordPress dans `functions.php`



1. l'indiquer à WordPress

→ enregistrer un emplacement de widget

```
function creer_widgets() {  
    register_sidebar(array(  
        'name' => 'Ma nouvelle zone de widget',  
        'id' => 'zone-widget',  
        'before_widget' => '<div class="zone-widget">',  
        'after_widget' => '</div>',  
        'before_title' => '<h2 class="zone-widget-title">',  
        'after_title' => '</h2>',  
    ));  
}  
  
add_action('widgets_init', 'creer_widgets');
```

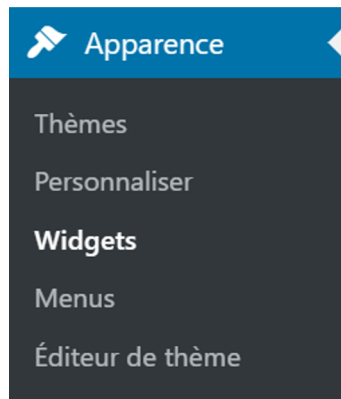


Vous pouvez utiliser plus d'une fois `register_sidebar()` afin d'avoir plusieurs emplacements de widget

```
register_sidebar(array(  
    // nom de l'emplacement qui apparaîtra dans l'interface administrateur  
    'name' => 'Ma nouvelle zone de widget',  
  
    // identifiant unique de l'emplacement widget  
    'id' => 'zone-widget',  
  
    // balise HTML qui ouvrira votre emplacement, ajoutez-y une classe qui vous sera utile pour le styler  
    'before_widget' => '<div class="zone-widget">',  
  
    // fermer la balise  
    'after_widget' => '</div>',  
  
    // balise pour le titre du widget (<h2>, <h3>, <h4>), ajoutez-y une classe qui vous sera utile pour le styler  
    'before_title' => '<h2 class="zone-widget-title">',  
  
    // fermer la balise du titre  
    'after_title' => '</h2>',  
));  
}
```

2. création et assignation d'un widget

il faut ensuite aller créer notre premier widget dans l'interface d'administration

A screenshot showing two stacked widget configuration forms. The top form is titled 'Articles récents' and contains: a title field with 'Articles récents', a 'Nombre d'articles à afficher' field with the value '3', a checked checkbox for 'Afficher la date de la publication', a 'Supprimer' link, and an 'Enregistrer' button. The bottom form is titled 'Catégories' and contains: a title field with 'Mots clés', three checkboxes ('Afficher comme liste déroulante' is unchecked, 'Afficher le nombre d'articles' is checked, and 'Afficher la hiérarchie' is unchecked), a 'Supprimer' link, and an 'Enregistrer' button.

3. sélection d'un emplacement

Le menu widget est maintenant disponible, mais rien ne s'affiche dans notre page.

Ne vous inquiétez pas, c'est normal! Nous devons maintenant lui indiquer l'endroit où les widgets devraient s'afficher. Nous appelons ça une **zone d'emplacement**.

3. sélection d'un emplacement (bis)

Pour déterminer une zone dans votre thème, vous devrez mettre le code suivant à l'endroit désiré dans votre gabarit :

```
<?php if (is_active_sidebar('zone-widget')) { ?>
    <?php dynamic_sidebar('zone-widget'); ?>
<?php } ?>
```

Petit conseil pour plus facilement ajuster le CSS : ajoutez un div pour englober la zone de widget.

```
<?php if (is_active_sidebar('zone-widget')) { ?>
    <div class="widget_header">
        <?php dynamic_sidebar('zone-widget'); ?>
    </div>
<?php } ?>
```

les types personnalisés

(custom post types, CPT)

retour sur « the loop »

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

```
<!-- s'il y a au moins un article à afficher -->
```

```
<?php if (have_posts()) : ?>
```

```
    <!-- tant qu'il y a un nouvel article à afficher, roule la boucle -->
```

```
    <?php while (have_posts()) : the_post(); ?>
```

```
        <!-- structure html pour afficher le contenu -->
```

```
    <?php endwhile; ?>
```

```
<?php else: ?>
```

```
    <!-- s'il n'y a aucun article à afficher -->
```

```
<?php endif; ?>
```

La boucle récupère automatiquement les « **posts** » correspondant au gabarit sélectionné :

- **page.php** : page sélectionnée
- **post.php** : article sélectionné
- **category.php** : tous les articles de la catégorie sélectionnée

Les fonctions comme **the_title()** ou **the_content()** doivent être utilisées à l'intérieur de la boucle **while()**.

Il est possible de modifier ou de créer une boucle pour récupérer du contenu plus spécifique ou personnalisé.



**créer les types
personnalisés**

identique à ce qu'on a déjà vu

**afficher les types
personnalisés**

nouvelle page

créons une nouvelle page **partenaires** avec un gabarit personnalisé (**page-partenaires.php**) basé sur **index.php** dans laquelle nous allons afficher la liste des partenaires.

le nouveau gabarit

en se basant sur index.php, remplaçons « la boucle » afin d'utiliser `query_posts()`.

```
<?php
    query_posts(array(
        'post_type' => 'cw4_partenaires',
        'post_status' => 'publish',
        'showposts' => 10
    ));
?>

<?php while (have_posts()) : the_post(); ?>
    <h2><?php the_title(); ?></h2>
    <?php the_content(); ?>
<?php endwhile; ?>

<!-- Réinitialiser la boucle d'origine -->
<?php wp_reset_query(); ?>
```

les partenaires sont là

mais le contenu de la page
Partenaires a disparu.. 🤖

Montréal en lumière

[Accueil](#) [Information](#) [À Propos](#) [Shenzen](#) [Partenaires](#)

À ne pas manquer!

Cette année la ville de Shenzen à l'honneur

[Apprenez-en plus...](#)



Partenaires Or

Bell



Bell lorem ipsum sit dolor amet.

Partenaires Argent

Casino de Montréal



Le Casino de Montréal lorem ipsum sit dolor amet.

Le lait



Le lait lorem ipsum sit dolor amet.

RBC



RBC lorem ipsum sit dolor amet.

Les à-côtés

Nouveaux spectacles

[Betty Bonifassi](#)
[Radio Radio](#)
[Mouse on the Keys](#)

Types d'événements

[Arts Numériques \(3\)](#)
[Exposition \(1\)](#)
[Musiques \(3\)](#)
[Partenaire argent \(3\)](#)
[Partenaire or \(1\)](#)
[Spectacles \(7\)](#)

combinons les « boucles »

afin d'accéder au contenu de la page en cours et la liste des partenaires

```
<div id="principal">
    <?php if( have_posts() ) : ?>
        <?php while (have_posts()) : the_post(); ?>
            <article <?php post_class(); ?> id="article"<?php the_ID(); ?>">
                <h3><?php the_title(); ?></h3>
                <?php the_post_thumbnail('vignette') ?>
                <?php the_excerpt(); ?>
                <a class="more-link" href="<?php the_permalink(); ?> ">Suite&hellip;</a>
            </article>
        <?php endwhile; ?>
    <?php endif ?>

    <?php
        query_posts(array(
            'post_type' => 'cw4_partenaires',
            'post_status' => 'publish',
            'showposts' => 10
        ));
    ?>
    <?php while (have_posts()) : the_post(); ?>
        <h2><?php the_title(); ?></h2>
        <?php the_content(); ?>
    <?php endwhile; ?>

    <?php wp_reset_query(); ?>
</div>
```

voilà!

tout devrait y être

À ne pas manquer!

Cette année la ville de Shenzen à l'honneur

[Apprenez-en plus...](#)



Partenaires

Nous voulons remercier nos partenaires pour leur précieuse collaboration à notre événement. Sans eux, ça ne serait tout simplement pas [...]

[Suite...](#)

Partenaires

Casino de Montréal



Le Casino de Montréal lorem ipsum sit dolor amet.

Bell



Bell lorem ipsum sit dolor amet.

RBC



RBC lorem ipsum sit dolor amet.

Le lait



Le lait lorem ipsum sit dolor amet.

Les à-côtés

Nouveaux spectacles

[Betty Bonifassi](#)
[Radio Radio](#)
[Mouse on the Keys](#)

Types d'événements

[Arts Numériques \(3\)](#)
[Exposition \(1\)](#)
[Musiques \(3\)](#)
[Partenaire argent \(3\)](#)
[Partenaire or \(1\)](#)
[Spectacles \(7\)](#)

concrètement, on veut regrouper les partenaires par catégories.

nous aurons à utiliser 3 boucles.

- boucle pour le contenu
- boucle pour les partenaires or
- boucle pour les partenaires argent

À ne pas manquer!

Cette année la ville de Shenzen à l'honneur

[Apprenez-en plus...](#)

Partenaires

Nous voulons remercier nos partenaires pour leur précieuse collaboration à notre événement. Sans eux, ça ne serait tout simplement pas [...]

[Suite...](#)

Partenaires Or

Bell



Bell lorem ipsum sit dolor amet.

Partenaires Argent

Casino de Montréal



Le Casino de Montréal lorem ipsum sit dolor amet.

Le lait



Le lait lorem ipsum sit dolor amet.

RBC



RBC lorem ipsum sit dolor amet.

Les à-côtés

Nouveaux spectacles

[Betty Bonifassi](#)
[Radio Radio](#)
[Mouse on the Keys](#)

Types d'événements

[Arts Numériques \(3\)](#)
[Exposition \(1\)](#)
[Musiques \(3\)](#)
[Partenaire argent \(3\)](#)
[Partenaire or \(1\)](#)
[Spectacles \(7\)](#)

vous pourriez mettre le contenu de la page après celui du CPT si vous voulez.

```
<?php
    query_posts(array(
        'post_type' => 'cw4_partenaires',
        'post_status' => 'publish',
        'showposts' => 10
    ));
?>
<?php while (have_posts()) : the_post(); ?>
    <h2><?php the_title(); ?></h2>
    <?php the_post_thumbnail('medium'); ?>
    <?php the_content(); ?>
<?php endwhile; ?>

<?php wp_reset_query(); ?>

<?php if( have_posts()) : ?>
    <?php while (have_posts()) : the_post(); ?>
        <article <?php post_class(); ?>>
            <h3><?php the_title(); ?></h3>
            <?php the_post_thumbnail('medium') ?>
            <?php the_excerpt(); ?>
        </article>
    <?php endwhile; ?>
<?php endif ?>
```



par défaut, **query_posts()** remplace la requête principale d'une page, si vous omettez **wp_reset_query()** après votre CPT, le contenu de la page principale ne sera pas disponible

définition de `wp_reset_query();`

- `wp_reset_query()` restaure les données `$wp_query` et global post à la requête principale d'origine.
- La requête d'origine est celle passée à la page par WordPress selon la page affichée.
- Cette fonction devrait être appelée après la boucle qui affiche les résultats d'un **`query_posts()`**.

**les paramètres
des query_posts();**

les `query_posts()`;

ça permet de faire une requête SQL dans WordPress sur :

- author
- category
- tag
- post & page
- status
- order
- date
- etc.

en gros, ça permet d'aller chercher de l'information complémentaire à celle de la page en cours qui ne serait naturellement pas disponible dans votre gabarit.



filtrer par type d'articles avec query_posts();

nous avons vu les paramètres suivants déjà :

```
<?php
    query_posts(array(
        'post_type' => 'cw4_partenaires',
        'post_status' => 'publish',
        'showposts' => 10
    ));
?>
```

post_type
clé du type de contenu
personnalisé

post_status
visibilité de la
publication

showposts
nombre d'article que
devrait contenir la
requête

filtrer par catégories

☐ Titre

MODIFICATION RAPIDE

Titre

Slug

Date à h min

Mot de passe – OU – ☐ Privé

Catégories

- ☒ Partenaires
 - ☒ Partenaire argent
 - ☐ Partenaire bronze
 - ☐ Partenaire or
- ☐ Spectacles

on ajoute les catégories afin de les lier à notre contenu

```
<?php
    query_posts(array(
        'post_type' => 'cw4_partenaires',
        'category_name' => 'partenaire_argent',
        'post_status' => 'publish',
        'showposts' => 10
    ));
?>
```

on indique ensuite dans notre requête qu'on cherche le contenu lié à cette catégorie

un exemple avec une catégorie

```
<h2>Partenaires Or</h2>
<?php
    query_posts(array(
        'post_type' => 'cw4_partenaires',
        'category_name' => 'partenaire_or',
        'post_status' => 'publish',
        'showposts' => 10
    ));
?>
<?php while (have_posts()) : the_post(); ?>
    <h2><?php the_title(); ?></h2>
    <?php the_content(); ?>
<?php endwhile; ?>

<?php wp_reset_query(); ?>
```

Partenaires

Nous voulons remercier nos partenaires pour leur précieuse collaboration à notre événement. Sans eux, ça ne serait tout simplement pas [...]

[Suite...](#)

Partenaires Or

Bell

The logo for Bell, consisting of the word "Bell" in a bold, blue, sans-serif font.

Bell lorem ipsum sit dolor amet.

on ajoute maintenant une autre catégorie

```
<h2>Partenaires Argent</h2>
<?php
    query_posts(array(
        'post_type' => 'cw4_partenaires',
        'category_name' => 'partenaire_argent',
        'post_status' => 'publish',
        'showposts' => 10
    ));
?>
<?php if (have_posts()): ?>
    <?php while (have_posts()) : the_post(); ?>
        <article <?php post_class(); ?>
            id="post-<?php the_ID(); ?>"
            <h3><?php the_title(); ?></h3>
            <?php the_post_thumbnail('medium'); ?>
            <?php the_content(); ?>
        </article>
    <?php endwhile; ?>
<?php endif; ?>

<?php wp_reset_query(); ?>
```

Partenaires

Nous voulons remercier nos partenaires pour leur précieuse collaboration à notre événement. Sans eux, ça ne serait tout simplement pas [...]

[Suite...](#)

Partenaires Or

Bell



Bell lorem ipsum sit dolor amet.

Partenaires Argent

Casino de Montréal



Le Casino de Montréal lorem ipsum sit dolor amet.

RBC



RBC lorem ipsum sit dolor amet.

Le lait



Le lait lorem ipsum sit dolor amet.

on ajoute un tri pour les partenaires argent

```
<h2>Partenaires Argent</h2>
<?php
    query_posts(array(
        'post_type' => 'cw4_partenaires',
        'category_name' => 'partenaire_argent',
        'orderby' => 'title',
        'order' => 'ASC',
        'post_status' => 'publish',
        'showposts' => 10
    ));
?>
<?php if (have_posts()): ?>
    <?php while (have_posts()) : the_post(); ?>
        <article <?php post_class(); ?>
            id="post-<?php the_ID(); ?>"
            <h3><?php the_title(); ?></h3>
            <?php the_post_thumbnail('medium'); ?>
            <?php the_content(); ?>
        </article>
    <?php endwhile; ?>
<?php endif; ?>

<?php wp_reset_query(); ?>
```

Partenaires

Nous voulons remercier nos partenaires pour leur précieuse collaboration à notre événement. Sans eux, ça ne serait tout simplement pas [...]

[Suite...](#)

Partenaires Or

Bell



Bell lorem ipsum sit dolor amet.

Partenaires Argent

Casino de Montréal



Le Casino de Montréal lorem ipsum sit dolor amet.

Le lait



Le lait lorem ipsum sit dolor amet.

RBC



RBC lorem ipsum sit dolor amet.



la fin