

**WordPress**

# **votre propre thème**

**on est rendu là**

Semaine 5

PRÉPARÉ PAR:  
**Bernard Imbert**  
**Jean-François Leblanc**  
Adapté par Mathieu Dupont  
Pour le cours 420-964-EM



# kahoot!

à vous de jouer

**un thème  
personnalisé**

FURROW



# DIG DEEP

Site of the Day

[furrow.studio](http://furrow.studio)



# ce projets services



02

03

14  
15  
16





We are a house  
of creativity



**tout est  
possible  
avec  
WordPress**

n'importe quel type de site peut fonctionner avec le moteur de WordPress  
il ne suffit que d'être original

---

**AWWWARDS®**

[sites utilisant WordPress](#)



[vitrine officielle de WordPress](#)

# **les bases**

parce qu'il faut bien commencer  
quelque part

## Comment on l'active?

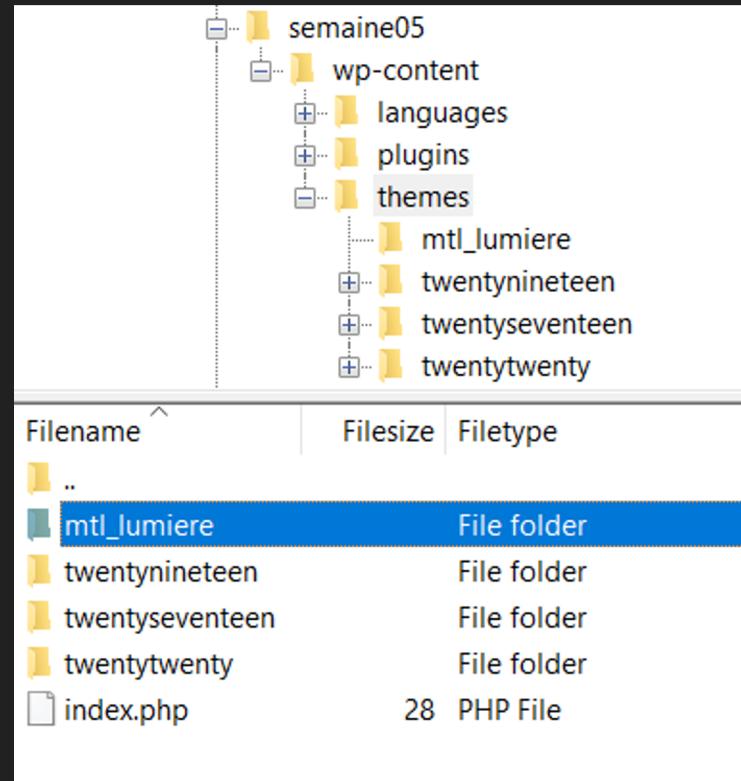
- lorsqu'on crée un thème personnalisé, on ne peut pas l'installer directement par l'interface administrateur de WordPress
- il faut le téléverser au bon endroit à l'aide d'un client FTP (FileZilla)
- vous pourrez ensuite activer votre thème à partir de l'interface administrateur

# pour créer un nouveau thème

il suffit de créer un sous-dossier dans **wp-content/themes/** et d'y placer les 3 fichiers magiques :

- **index.php**
- **style.css**
- **templates/index.html**

(le fichier **index.html** doit se trouver dans un sous-dossier **templates**)



The screenshot shows a file explorer interface with two panes. The left pane displays a hierarchical tree view of a directory structure. The root folder is 'semaine05', which contains 'wp-content'. 'wp-content' contains 'languages', 'plugins', and 'themes'. 'themes' contains four subfolders: 'mtl\_lumiere', 'twentynineteen', 'twentyseventeen', and 'twentytwenty'. The right pane shows a list of files and folders from the 'mtl\_lumiere' folder. The columns are 'Filename', 'Filesize', and 'Filetype'. The list includes '..' (File folder), 'mtl\_lumiere' (File folder, highlighted with a blue selection bar), 'twentynineteen' (File folder), 'twentyseventeen' (File folder), 'twentytwenty' (File folder), and 'index.php' (PHP File, with a file size of 28).

| Filename        | Filesize | Filetype    |
|-----------------|----------|-------------|
| ..              |          |             |
| mtl_lumiere     |          | File folder |
| twentynineteen  |          | File folder |
| twentyseventeen |          | File folder |
| twentytwenty    |          | File folder |
| index.php       | 28       | PHP File    |

# capture d'écran de votre thème

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

- index.php
- style.css
- templates/index.html
- **screenshot.png**



Twenty Twenty-Two Version : 1.0

Par [the WordPress team](#)

[Activer les mises à jour auto](#)

Built on a solidly designed foundation, Twenty Twenty-Two embraces the idea that everyone deserves a truly unique website. The theme's subtle styles are inspired by the diversity and versatility of birds: its typography is lightweight yet strong, its color palette is drawn from nature, and its layout elements sit gently on the page. The true richness of Twenty Twenty-Two lies in its opportunity for customization. The theme is built to take advantage of the Full Site Editing features introduced in WordPress 5.9, which means that colors, typography, and the layout of every single page on your site can be customized to suit your vision. It also includes dozens of block patterns, opening the door to a wide range of professionally designed layouts in just a few clicks. Whether you're building a single-page website, a blog, a business website, or a portfolio, Twenty Twenty-Two will help you create a site that is uniquely yours.

# fichier style.css

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

```
/*
Theme Name: Twenty Twenty-Two
Author: the WordPress team
Author URI: http://wordpress.org/
Description: Built on a solidly designed...
Version: 1.0
Text Domain: twentytwentytwo
Tags: one-column, custom-colors, ...
*/
```



Version

Theme Name



About Birds All Posts

*The Hatchery: a blog  
about adventures in  
bird watching.*

Description



Tags

Author

Twenty Twenty-Two Version : 1.0

Par [the WordPress team](#)

[Activer les mises à jour auto](#)

Built on a solidly designed foundation, Twenty Twenty-Two embraces the idea that everyone deserves a truly unique website. The theme's subtle styles are inspired by the diversity and versatility of birds: its typography is lightweight yet strong, its color palette is drawn from nature, and its layout elements sit gently on the page. The true richness of Twenty Twenty-Two lies in its opportunity for customization. The theme is built to take advantage of the Full Site Editing features introduced in WordPress 5.9, which means that colors, typography, and the layout of every single page on your site can be customized to suit your vision. It also includes dozens of block patterns, opening the door to a wide range of professionally designed layouts in just a few clicks. Whether you're building a single-page website, a blog, a business website, or a portfolio, Twenty Twenty-Two will help you create a site that is uniquely yours.

Étiquettes : Une colonne, Couleurs personnalisées, custom-menu, Logo personnalisé, Éditeur de style, Images mises en avant, Modification complète du

Activer

Supprimer

# le fichier **functions.php**

# 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 charger des scripts, par exemple de CSS
- 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.



Code à insérer dans **functions.php** pour inclure le CSS du fichier **style.css** se trouvant à la racine du thème :

```
function mtl_theme_scripts() {
    wp_enqueue_style( 'style', get_stylesheet_uri() );
}
add_action( 'wp_enqueue_scripts', 'mtl_theme_scripts' );
```

## **Vous avez créé votre premier thème**

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

- le fichier index.html contient un mélange de code HTML et de commentaires HTML pour insérer des blocs. C'est de cette façon que sont construits les gabarits WordPress

# récupération du contenu

on l'a entré dans le CMS,  
faudrait bien qu'ça serve!

# Comment récupérer et intégrer les contenus de WordPress?

- il faut utiliser un bloc de type « **query** » qui remplace « **la boucle** »
- par défaut, la boucle affiche les 10 derniers articles. Cela peut-être modifié dans la section **Réglages > lecture**

# le bloc query

```
<!-- wp:query -->
<div class="wp-block-query">
    <!-- wp:post-template -->
        <!-- wp:post-title /-->
        <!-- wp:post-featured-image {"isLink":true} /-->
        <!-- wp:post-excerpt /-->
    <!-- /wp:post-template -->
</div>
<!-- /wp:query -->
```

# plusieurs options possibles dans un « query »

- <!-- wp:post-excerpt /--> **résumé**
- <!-- wp:post-content /--> **contenu**
- <!-- wp:post-featured-image /--> **image à la une**
- <!-- wp:post-terms {"term":"category"} /--> **catégorie(s)**
- <!-- wp:post-terms {"term":"post\_tag"} /--> **mot(s)-clef(s)**
- <!-- wp:post-date /--> **date de publication**
- <!-- wp:post-author /--> **auteur**
- <!-- wp:post-title /--> **titre**

# **les gabarits WordPress**

il y en a une panoplie...

# c'est quoi un gabarit WordPress?

Le contenu de notre site Web risque d'avoir des mises en page différentes d'une section à l'autre. Les gabarits sont là pour ça.

Ainsi, une **page** pourra être présentée différemment d'un **article** par exemple.

Pour chaque type de section, WordPress tentera par défaut de trouver le gabarit le mieux adapté selon un arbre de décision.

# les principaux gabarits

| TYPE DE PAGE / SECTION            | URL (PAR DÉFAUT)                 | GABARIT UTILISÉ |
|-----------------------------------|----------------------------------|-----------------|
| accueil                           | /                                | home.html       |
| archives<br>(année, mois ou jour) | /2020<br>/2020/01<br>/2020/01/22 | date.html       |
| catégorie                         | /category/geek                   | category.html   |
| mot-clé                           | /tag/iphone                      | tag.html        |
| résultat de recherche             | ?s=cadeau                        | search.html     |
| post                              | /2011/12/25/le-dec-tim           | single.html     |
| page                              | /a-propos                        | page.html       |
| 404                               | /page-qui-nexiste-pas            | 404.html        |



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

WordPress vérifie l'existence des gabarits de gauche à droite dans le tableau de la page précédente.

Si un gabarit n'existe pas dans le dossier de thème, il passe au suivant et ainsi de suite.

Vous n'êtes pas obligés de tous les créer car il y aura toujours le fichier **index.html** qui est utilisé par défaut au bout de la chaîne. Ce dernier est donc obligatoire.

## **front-page.html vs home.html ?**

- **front-page.html : assigné à la page d'accueil**, qu'il s'agisse d'une page statique ou de la liste des derniers articles
- **home.html : assigné à la page affichant la liste des derniers articles**, que ce soit la page d'accueil ou non  
(sauf si c'est la page d'accueil ET que **front-page.html existe**)

# un gabarit pour les pages : page.html

- Si on veut une mise en page différente pour les pages versus les articles, il suffit de créer un fichier `page.html` et d'ajuster le design à nos besoins.
- Si WordPress est configuré pour afficher une page statique comme page d'accueil (plutôt que la liste d'articles), le gabarit `front-page.html` sera assigné à la page en question
- pour créer un gabarit spécifique à une seule page :
  - `page-nom.html` (si le site utilise les liens optimisés pour le SEO)
  - `page-id.html` (si le site utilise les liens du type ?p=12)

# un gabarit pour plusieurs pages

- on peut créer des gabarits de page pouvant être utilisés par plusieurs pages (sélection manuelle)
- pour se faire, il suffit de créer un fichier HTML dans le dossier templates du thème, puis de l'assigner à une page par l'interface d'administration :

Attributs de page ^

Modèle :

|                   |
|-------------------|
| Modèle par défaut |
| Modèle par défaut |
| Mon gabarit       |
| (Aucun parent)    |



## **page-nom.html vs [nom\_gabarit].html?**

- **page-nom.html** est intéressant pour un site qui change peu. Un gros désavantage est que lorsque le nom de la page change, il faut changer le gabarit aussi
- **[nom\_gabarit].html** est intéressant car il s'adapte peu importe le contenu des pages. Comme il est configuré dans une page, on n'a rien à changer au niveau du gabarit pour qu'il continue de fonctionner normalement

la meilleure pratique?

**un gabarit qu'on peut  
choisir dans les réglages  
d'une page**

# À propos

Classique

## À propos du festival

MONTRÉAL EN LUMIÈRE, c'est un des plus grands festivals d'hiver au monde, une folie qui enregistre annuellement 1 300 000 visites de festivaliers venus vivre pleinement l'hiver montréalais par le biais d'une programmation inusitée, alliant arts de la scène, gastronomie et activités familiales extérieures gratuites, sans oublier une nuit complète de découvertes des plus originales !

Document Bloc

État et visibilité

Permalien

Image mise en avant

Définir l'image mise en avant

Discussion

Attributs de page

Modèle : Modèle par défaut

Page parente : (Aucun parent)

Ordre 0

# **les fichiers en inclusions**

copier/coller on aime ça...

- Certaines parties du code étant généralement les mêmes à travers les différentes pages d'un site, il est préférable de regrouper ces parties dans des fichiers séparés que l'on va inclure dans le fichier principal d'affichage des contenus.
- Cela évitera d'avoir à faire la même modification dans plusieurs fichiers et limitera les possibilités d'erreurs ou d'oubli.
- Lorsqu'on utilise une fonction d'inclusion, c'est comme si on indiquait de copier le code du fichier à inclure à l'endroit où la fonction est appelée dans le fichier principal.

# une structure réutilisable et modulaire

header.html

index.html

sidebar.html

footer.html

# les fichiers en inclusion

- en-tête : header.html
- pied de page : footer.html
- zone secondaire : sidebar.html
- etc.

# appeler les fichiers en inclusion

```
→ <!-- wp:template-part {"slug":"header"} /-->  
    header.html  
  
→ <!-- wp:template-part {"slug":"footer"} /-->  
    footer.html  
  
→ <!-- wp:template-part {"slug":"sidebar"} /-->  
    sidebar.html  
  
→ <!-- wp:template-part {"slug":"composante"} /-->  
    composante.html
```



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

## le fichier header.html

- ce fichier contiendra le contenu qui débutera chaque page
- en fait, il contient habituellement le bloc <**header**>

# header.html

```
<header>
    [ ...contenu... ]
</header>
```

## le fichier footer.html

- ce fichier contiendra le contenu qui terminera chaque page
- en fait, il contient habituellement le bloc <**footer**>

# footer.html

```
<footer>  
    ©2022, Montréal en lumières...  
</footer>
```

## le fichier sidebar.html

- ce fichier contiendra le contenu qui se situera dans une colonne latérale ou une région quelconque
- 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 contient habituellement un bloc `<aside>`

# sidebar.html

```
<aside>  
    [ . . . contenu . . . ]  
</aside>
```

# index.html révisé avec les inclusions

```
<!-- wp:template-part {"slug":"header"} -->

<!-- wp:query -->
<div class="wp-block-query">
    <!-- wp:post-template -->
        <!-- wp:post-title -->
        <!-- wp:post-featured-image {"isLink":true} -->
        <!-- wp:post-excerpt -->
    <!-- /wp:post-template -->
</div>
<!-- /wp:query -->

<!-- wp:template-part {"slug":"sidebar"} -->
<!-- wp:template-part {"slug":"footer"} -->
```



la fin

exercice s05

# créer son thème personnalisé

➔ instructions sur le site du cours dans semaine05 ➔