



M'Zlle Flo

Annexes

Codes

Michel Grillon 07-08 / 2024



Table des matières

<i>On débute avec du JavaScript et le fichier scripts.js :.....</i>	<i>3</i>
<i>style.css :.....</i>	<i>9</i>
<i>functions.php :.....</i>	<i>32</i>
<i>header.php :.....</i>	<i>60</i>
<i>Footer.php :.....</i>	<i>62</i>
<i>index.php :.....</i>	<i>64</i>
<i>page-full.php :.....</i>	<i>65</i>
<i>plugin-art-gallery.php :</i>	<i>66</i>
<i>plugin-advanced-contact.php :</i>	<i>73</i>



On débute avec du JavaScript et le fichier scripts.js :

// Gestion de la recherche

```
jQuery(document).ready(function($) {  
    // Lorsque l'icône de recherche est cliquée, affiche ou masque le formulaire de recherche  
    $('.search-icon').click(function() {  
        $('#searchform').toggleClass('visible');  
    });  
});
```

```
document.addEventListener("DOMContentLoaded", function () {
```

// Gestion du menu mobile

```
    const mobileMenuIcon = document.getElementById("mobile-menu-icon"); // Récupère l'icône du  
    menu mobile
```

```
    const menuPrincipal = document.getElementById("menuPrincipal"); // Récupère le menu principal
```

// Lorsque l'icône de menu mobile est cliquée, le menu se déplie ou se replie

```
    mobileMenuIcon.addEventListener("click", function () {  
        menuPrincipal.classList.toggle("open"); // Ajoute ou retire la classe "open" pour déplier ou  
        replier le menu  
    });
```

// JavaScript pour la gestion des sous-menus en version mobile en lieu et place du "hover".

```
    const menuItems = document.querySelectorAll('#menuPrincipal > ul > li'); // Sélectionne tous les  
    éléments de menu
```

```
    menuItems.forEach(function (item) {
```

// Ajoute un événement de clic pour chaque élément de menu

```
    item.addEventListener('click', function (event) {  
        event.stopPropagation(); // Empêche la propagation de l'événement  
        const submenu = this.querySelector('ul'); // Sélectionne le sous-menu associé  
        if (submenu) {  
            submenu.classList.toggle('show'); // Affiche ou masque le sous-menu  
        }  
    });
```



// Ajoute un événement de survol pour chaque élément de menu

```
item.addEventListener('mouseover', function () {  
    const submenu = this.querySelector('ul'); // Sélectionne le sous-menu associé  
    if (submenu) {  
        submenu.classList.add('show'); // Affiche le sous-menu  
    }  
});
```

// Ajoute un événement de sortie de survol pour chaque élément de menu

```
item.addEventListener('mouseout', function () {  
    const submenu = this.querySelector('ul'); // Sélectionne le sous-menu associé  
    if (submenu) {  
        submenu.classList.remove('show'); // Masque le sous-menu  
    }  
});  
});
```

// Ferme tous les sous-menus lorsqu'on clique à l'extérieur du menu

```
document.addEventListener('click', function (event) {  
    if (!event.target.closest('#menuPrincipal')) {  
        const submenus = document.querySelectorAll('#menuPrincipal ul.show'); // Sélectionne tous  
        les sous-menus affichés  
        submenus.forEach(function (submenu) {  
            submenu.classList.remove('show'); // Masque les sous-menus  
        });  
    }  
});
```

// Ajout le code pour la flèche de retour en haut de la page

```
const retourHaut = document.getElementById("retour-haut"); // Récupère l'élément de la flèche de  
retour en haut
```

// Écoutez le défilement de la page

```
window.addEventListener("scroll", function() {  
    if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {  
        retourHaut.style.display = "block"; // Affiche la flèche de retour en haut  
    } else {  
        retourHaut.style.display = "none"; // Masque la flèche de retour en haut  
    }  
});
```

// Ajout un gestionnaire d'événements pour le clic sur la flèche

```
retourHaut.addEventListener("click", function() {
```



```
document.body.scrollTop = 0; // Pour les navigateurs Chrome, Safari
document.documentElement.scrollTop = 0; // Pour les navigateurs Firefox, IE, Opera
});

// Initialisation de la lightbox
lightbox.option({
  'resizeDuration': 200, // Durée de l'animation de redimensionnement
  'wrapAround': true // Permet de boucler les images
});

// Filtrage des œuvres d'art
const filterCategory = document.getElementById('filter-category'); // Récupère le sélecteur de
catégorie
if (filterCategory) {
  filterCategory.addEventListener('change', function () {
    const category = this.value; // Récupère la catégorie sélectionnée
    const items = document.querySelectorAll('.art-gallery-item'); // Récupère tous les éléments de
la galerie
    items.forEach(item => {
      if (category === '' || item.classList.contains(category)) {
        item.style.display = 'block'; // Affiche l'élément s'il correspond à la catégorie sélectionnée
      } else {
        item.style.display = 'none'; // Masque l'élément s'il ne correspond pas à la catégorie
sélectionnée
      }
    });
  });
}

// Animation des images au survol
const galleryItems = document.querySelectorAll('.art-gallery-item'); // Récupère tous les éléments
de la galerie
galleryItems.forEach(item => {
  // Ajoute un événement de survol pour chaque élément de la galerie
  item.addEventListener('mouseover', function () {
    this.style.transform = 'translateY(-10px) scale(1.05)'; // Anime l'élément en le déplaçant vers
le haut et en l'agrandissant
    this.style.boxShadow = '0 10px 20px rgba(0, 0, 0, 0.2)'; // Ajoute une ombre portée
  });
  // Ajoute un événement de sortie de survol pour chaque élément de la galerie
  item.addEventListener('mouseout', function () {
    this.style.transform = 'translateY(0) scale(1)'; // Réinitialise l'animation
    this.style.boxShadow = '0 2px 10px rgba(0, 0, 0, 0.1)'; // Réinitialise l'ombre portée
  });
});
```



```
});  
});  
  
// Gestion du chargement infini  
const loadMoreButton = document.getElementById('load-more'); // Récupère le bouton de  
chargement infini  
if (loadMoreButton) {  
    loadMoreButton.addEventListener('click', function () {  
        const nextPage = parseInt(this.getAttribute('data-next-page'), 10); // Récupère le numéro de la  
prochaine page  
        const maxPages = parseInt(this.getAttribute('data-max-pages'), 10); // Récupère le nombre  
maximal de pages  
        const selectedCategory = document.getElementById('filter-category').value; // Récupère la  
catégorie sélectionnée  
  
        if (nextPage <= maxPages) {  
  
            fetch(`${my_ajax_object.ajax_url}?action=load_more_artworks&page=${nextPage}&category=${selectedCategory}`, {  
                method: 'POST',  
            })  
                .then(response => response.text()) // Convertit la réponse en texte  
                .then(data => {  
                    const parser = new DOMParser(); // Crée un nouveau parseur DOM  
                    const doc = parser.parseFromString(data, 'text/html'); // Parse le texte en un document  
HTML  
                    const newItems = doc.querySelectorAll('.art-gallery-item'); // Récupère les nouveaux  
éléments de la galerie  
                    newItems.forEach(item => {  
                        document.querySelector('.art-gallery').appendChild(item); // Ajoute les nouveaux  
éléments à la galerie  
                    });  
  
                    // Réinitialiser les événements pour les nouveaux éléments  
                    newItems.forEach(item => {  
                        item.addEventListener('mouseover', function () {  
                            this.style.transform = 'translateY(-10px) scale(1.05)';  
                            this.style.boxShadow = '0 10px 20px rgba(0, 0, 0, 0.2)';  
                        });  
                        item.addEventListener('mouseout', function () {  
                            this.style.transform = 'translateY(0) scale(1)';  
                            this.style.boxShadow = '0 2px 10px rgba(0, 0, 0, 0.1)';  
                        });  
                    });  
                });  
            });  
        }  
    });  
}
```



```
});

// Mettre à jour la lightbox pour les nouveaux éléments
lightbox.init();

loadMoreButton.setAttribute('data-next-page', nextPage + 1); // Met à jour le numéro de
la prochaine page
if (nextPage + 1 > maxPages) {
    loadMoreButton.style.display = 'none'; // Masque le bouton si toutes les pages ont été
chargées
}
});
} else {
    loadMoreButton.style.display = 'none'; // Masque le bouton si toutes les pages ont été
chargées
}
});
}

// Ajout du filtrage par catégorie
if (filterCategory) {
    filterCategory.addEventListener('change', function () {
        const selectedCategory = this.value; // Récupère la catégorie sélectionnée

        fetch(`${my_ajax_object.ajax_url}?action=load_more_artworks&page=1&category=${selectedCategor
y}`, {
            method: 'POST',
        })
        .then(response => response.text()) // Convertit la réponse en texte
        .then(data => {
            const parser = new DOMParser(); // Crée un nouveau parseur DOM
            const doc = parser.parseFromString(data, 'text/html'); // Parse le texte en un document
HTML

            const newItems = doc.querySelectorAll('.art-gallery-item'); // Récupère les nouveaux
éléments de la galerie
            const artGallery = document.querySelector('.art-gallery'); // Récupère la galerie d'art
            artGallery.innerHTML = ""; // Réinitialise le contenu de la galerie
            newItems.forEach(item => {
                artGallery.appendChild(item); // Ajoute les nouveaux éléments à la galerie
            });

            // Réinitialiser les événements pour les nouveaux éléments
            newItems.forEach(item => {
```



```
item.addEventListener('mouseover', function () {
    this.style.transform = 'translateY(-10px) scale(1.05)';
    this.style.boxShadow = '0 10px 20px rgba(0, 0, 0, 0.2)';
});
item.addEventListener('mouseout', function () {
    this.style.transform = 'translateY(0) scale(1)';
    this.style.boxShadow = '0 2px 10px rgba(0, 0, 0, 0.1)';
});
});

// Mettre à jour la lightbox pour les nouveaux éléments
lightbox.init();

loadMoreButton.setAttribute('data-next-page', 2); // Met à jour le numéro de la prochaine
page
loadMoreButton.style.display = newItems.length ? 'block' : 'none'; // Affiche ou masque le
bouton en fonction du nombre d'éléments
});
});
}
});

// JavaScript personnalisé pour la galerie d'art
jQuery(document).ready(function($) {
    // Lorsque l'élément de la galerie d'art est cliqué, redirige vers la page du produit associé
    $('.artwork-item').on('click', function() {
        var productId = $(this).data('product-id'); // Récupère l'ID du produit associé à l'œuvre d'art
        if (productId) {
            window.location.href = '/product/' + productId; // Redirige vers la page du produit
        }
    });
});
});
```




style.css :

Il est à noter que :

 Service de validation CSS du W3C
Résultats de la validation W3C CSS de https://cefii-developpements.fr/michel1385/Projet_final/mzlleflo/accueil/ (CSS niveau 3 + SVG)

Aller à: [Avertissements \(87\)](#) [CSS valide](#)

Résultats de la validation W3C CSS de https://cefii-developpements.fr/michel1385/Projet_final/mzlleflo/accueil/ (CSS niveau 3 + SVG)

Félicitations ! Aucune erreur trouvée.
Ce document est valide conformément à la recommandation [CSS niveau 3 + SVG](#) !

Le site est valide conformément à la recommandation CSS niveau 3 + SVG !

(les avertissement ne concernent pas mes fichiers...).

Retour a style.css :

```
/*
Theme name: MZLLFlo Thème
Theme URI: http://example.com/
Author: M'ZLLFlo et Michel Grillon
Author URI: https://www.facebook.com/mzlleflo.deco
Description: Thème pour MZLLFlo avec menu déroulant, deux zones de widgets, une à droite et une
en bas. Style épuré et artistique.
Version: 1.0
License: GNU General Public License v2 or later License URI: http://www.gnu.org/licenses/gpl-
2.0.html
License URI: http://www.gnu.org/licenses/gpl-2.0.html
Tags: light, two-columns, responsive-layout, etc.
Text Domain: mzlleflo
*/

/*=====
=====*/

@import url('https://fonts.googleapis.com/css2?family=Great+Vibes&display=swap');
@import url('https://fonts.googleapis.com/css2?family=Send+Flowers&display=swap');
@import
url('https://fonts.googleapis.com/css2?family=Playwrite+NO:wght@100..400&display=swap');

/* Style pour la police Great Vibes */
.great-vibes-regular {
    font-family: "Great Vibes", cursive;
```



```
font-weight: 400;
font-style: normal;
}
```

```
/* Style pour la police Send Flowers */
.send-flowers-regular {
    font-family: "Send Flowers", cursive;
    font-weight: 400;
    font-style: normal;
}
```

```
/* Style pour la police Playwrite NO */
.playwrite-no {
    font-family: "Playwrite NO", cursive;
    /*font-optical-sizing: auto;*/
    font-weight: 400;
    font-style: normal;
}
```

```
.wc-block-grid__product-add-to-cart.wp-block-button .wp-block-button__link,
.button.product_type_simple.add_to_cart_button.ajax_add_to_cart,
button.product_type_simple.add_to_cart_button.ajax_add_to_cart.loading,
button.product_type_variable.add_to_cart_button,
a.button.product_type_variable.add_to_cart_button,
button.product_type_external,
button.product_type_grouped,
li#tab-title-description.description_tab.active,
li#tab-title-additional_information.additional_information_tab,
li#tab-title-reviews.reviews_tab,
button.single_add_to_cart_button.button.alt,
.woocommerce a.remove,
p.woocommerce-store-notice.demo_store{
    font-family: "Playwrite NO", cursive;
    font-weight: bold !important;
    font-size: 0.8rem !important;
    background-color: #E9E6ED;
    /*background-color: #DCD7E3;*/
    color: black;
    padding: 0.5em 1em;
    border-radius: 10px;
    word-break: break-word;
    white-space: normal;
    margin-right: auto !important;
```



```
margin-left: auto !important;
display: inline-flex;
justify-content: center;
text-align: center;
}

.wc-block-grid__product-add-to-cart.wp-block-button .wp-block-button__link,
.button.product_type_simple.add_to_cart_button.ajax_add_to_cart,
button.product_type_simple.add_to_cart_button.ajax_add_to_cart.loading,
button.product_type_variable.add_to_cart_button,
button.product_type_external,
button.product_type_grouped,
li#tab-title-description.description_tab.active,
li#tab-title-additional_information.additional_information_tab,
li#tab-title-reviews.reviews_tab,
button.single_add_to_cart_button.button.alt,
p.woocommerce-store-notice.demo_store:hover {
    font-family: "Playwrite NO", cursive;
    font-weight: bold !important;
    font-size: 0.8rem !important;
    background-color: #DCD7E3;
    color: black;
    padding: 0.5em 1em;
    border-radius: 10px;
    word-break: break-word;
    white-space: normal;
    margin-right: auto !important;
    margin-left: auto !important;
    display: inline-flex;
    justify-content: center;
    text-align: center;
}

/*=====*/

/* Style pour le corps de la page */
body {
    max-width: 80%;
    margin: auto;
}

/* Style pour les images */
img {
```



```
max-width: 100%;
height: auto;
}

/* Style pour les images mises en avant */
.post-thumbnail img {
max-width: 100%;
height: auto;
}

/* Style pour les images dans le contenu */
.entry-content img {
max-width: 100%;
height: auto;
}

/* Style pour le titre */
title {
font-family: "Great Vibes", cursive;
font-weight: 500;
font-style: normal;
text-decoration: none;
}

/* Style pour les titres de niveau 1 */
h1 {
font-family: "Great Vibes", cursive;
color: black;
}

/* Style pour les titres de niveau 2 à 6 */
h2,
h3,
h4,
h5,
h6 {
color: black;
font-family: "Send Flowers", cursive;
}

/* Style pour les liens */
a {
color: black;
```



```
text-decoration: none;
}

/* Style pour le conteneur principal */
#wrapper {
  max-width: 1280px;
  margin: 0 auto;
}

/* Style pour le titre de la catégorie */
.category-title {
  font-family: "Great Vibes", cursive;
  font-weight: 500;
  font-style: normal;
  text-decoration: none;
  color: black;
  text-align: center;
  margin-bottom: 20px;
}

/*=====*/

/* Style pour l'en-tête */
header {
  display: flex;
  justify-content: center;
  flex-direction: column;
  align-items: center;
  text-align: center;
}

/* Style pour les liens dans l'en-tête */
header a {
  display: flex;
  justify-content: center;
  align-items: center;
  min-width: 50%;
  max-width: 100%;
  text-decoration: none;
}

/* Style pour le logo */
#logo {
```



```
display: flex;
justify-content: center;
width: 100%;
margin-top: 10px;
}
```

```
/* Style pour l'image du logo */
```

```
#logo img {
    max-width: 100%;
    height: auto;
    margin: auto;
}
```

```
/* Style pour le titre du site */
```

```
#titre {
    width: 100%;
    text-align: center;
}
```

```
/* Style pour le menu principal */
```

```
#menuPrincipal {
    text-align: center;
    margin-top: 10px;
}
```

```
/* Style pour le menu principal */
```

```
#menuPrincipal #main-menu {
    display: inline-block;
}
```

```
/* Style pour l'en-tête du site */
```

```
#site-header {
    margin-bottom: 0.5em;
    padding: 0.5em;
    display: flex;
    flex-direction: column;
    align-items: center;
    text-align: center;
}
```

```
/* Style pour les liens dans l'en-tête du site */
```

```
#site-header a {
    display: flex;
```



```
justify-content: space-between;
justify-content: center;
align-items: center;
min-width: 50%;
}

/* Style pour le conteneur de recherche */
#recherche {
  display: flex;
  align-items: center;
  position: relative;
  /* Déplace l'icône en dessous du menu */
  margin-top: 10px;
  width: 100%;
  /* Aligne l'icône à l'extrême droite */
  justify-content: flex-end;
  box-shadow: 0px 8px 16px rgba(0, 0, 0, 0.2);
  /* Optionnel : pour une ombre portée */
}
```

```
/* Style pour le champ de recherche */
#searchform {
  display: none;
  position: absolute;
  /* Positionne le champ de recherche à gauche de l'icône */
  right: 100%;
  top: 0;
  transition: right 0.3s ease;
  /* Empêche le texte de se déplacer sur une nouvelle ligne */
  white-space: nowrap;
  box-shadow: 0px 8px 16px rgba(0, 0, 0, 0.2);
  /* Optionnel : pour une ombre portée */
}
```

```
/* Style pour le champ de recherche */
#recherche input {
  border-radius: 10px;
  padding-left: 10px;
  padding-right: 10px;
  height: 2em;
  border: none;
  box-shadow: 0px 8px 16px rgba(0, 0, 0, 0.2);
  /* Optionnel : pour une ombre portée */
}
```



```
}
```

```
/* Style pour afficher le champ de recherche visible */
```

```
#searchform.visible {
```

```
    display: block;
```

```
    right: 0;
```

```
}
```

```
/* Style pour le premier champ de recherche */
```

```
#recherche input:first-of-type {
```

```
    color: #ccc;
```

```
}
```

```
/* Style pour cacher le label du formulaire de recherche */
```

```
#searchform label {
```

```
    display: none;
```

```
}
```

```
/* Style pour l'icône de recherche */
```

```
.search-icon {
```

```
    cursor: pointer;
```

```
    margin-left: 10px;
```

```
    position: absolute;
```

```
    right: 0;
```

```
    top: 50%;
```

```
    transform: translateY(-50%);
```

```
    z-index: 10;
```

```
}
```

```
/* Style pour le menu */
```

```
nav#menuPrincipal {
```

```
    margin-bottom: 1em;
```

```
    padding-left: 0.5em;
```

```
    font-family: "Playwrite NO", cursive;
```

```
    font-weight: bold;
```

```
}
```

```
/* Style pour les listes dans le menu */
```

```
nav#menuPrincipal ul {
```

```
    list-style-type: none;
```

```
    padding: 0;
```

```
    font-family: "Playwrite NO", cursive;
```

```
    font-weight: bold;
```




```
display: flex;
align-items: center;
}
```

```
/* Style pour les éléments de liste dans le menu */
```

```
nav#menuPrincipal ul li {
display: block;
margin-bottom: .5em;
position: relative;
font-family: "Playwrite NO", cursive;
font-weight: bold;
}
```

```
/* Style pour les liens dans les éléments de liste du menu */
```

```
nav#menuPrincipal ul li a {
display: block;
text-decoration: none;
font-family: "Playwrite NO", cursive;
font-weight: bold;;
background-color: #fff;
color: black;
box-shadow: 0px 8px 16px rgba(0, 0, 0, 0.2);
/* Optionnel : pour une ombre portée */
padding: 0.5em 1em;
border-radius: 10px;
text-align: center;
}
```

```
/* Style pour les sous-menus */
```

```
nav#menuPrincipal ul li ul li {
display: block;
width: 100%;
text-align: center;
border-top: 1px solid transparent;
padding: 0;
}
```

```
/* Style pour les liens dans les sous-menus */
```

```
nav#menuPrincipal ul li ul li a {
padding: 0.5em 0;
}
```

```
/* Style pour les sous-menus */
```



```
nav#menuPrincipal ul li ul {  
    /* Masquer les sous-menus par défaut */  
    display: none;  
    /* Positionner absolument pour apparaître en dessous du menu principal */  
    position: absolute;  
    left: 0;  
    /* Placer le sous-menu juste en dessous du menu principal */  
    top: 100%;  
    /* Assurer que le sous-menu est au-dessus des autres éléments */  
    z-index: 1000;  
    flex-direction: column;  
    text-align: center;  
    list-style-type: none;  
    padding: 0.05em;  
}  
  
/* Style pour les sous-menus au survol */  
nav#menuPrincipal ul li: hover > ul {  
    display: block;  
}  
  
/* Cache les sous-menus par défaut */  
ul.sub-menu {  
    display: none;  
    position: absolute;  
    border-radius: 10px;  
    width: 4.2em;  
    text-align: center;  
    list-style-type: none;  
}  
  
/* Affiche le sous-menu au survol du parent */  
li.menu-item: hover > ul.sub-menu {  
    display: block;  
}  
  
/* Optionnel : Assure que le sous-menu reste visible lorsqu'il est survolé */  
li.menu-item ul.sub-menu: hover {  
    display: block;  
}  
  
/* Style pour l'icône du menu mobile */  
#mobile-menu-icon {
```



```
display: none;
cursor: pointer;
font-size: 1.5em;
position: absolute;
top: 5px;
right: 0;
}
```

```
/* Style pour l'icône du menu mobile */
#mobile-menu-icon i {
    color: #000;
}
```

```
/* Style pour le menu principal ouvert */
#menuPrincipal.open {
    display: block;
    justify-content: right;
    width: 50%;
    position: absolute;
    top: 5px;
    right: 0;
    background-color: #fff;
    color: #000000;
    z-index: 1000;
    opacity: 0.9;
    transition: opacity 1s ease-in-out;
/* Ombre légère */
    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);
}
```

```
/* Style pour le bouton de basculement du menu */
.toggle-nav {
    display: none !important;
}
```

```
/* Style pour la gestion des sous-menus en version mobile à la place du hover */
nav#menuPrincipal ul li ul.show {
    display: block;
}
```

```
/* Style pour les sous-menus actifs */
nav#menuPrincipal ul li ul.active {
    display: block;
```



}

/*=====*/

/ Style pour le contenu principal */*

```
body.main-container main {  
    display: flex;  
    justify-content: space-between;  
    font-family: "Playwrite NO", cursive;  
}
```

/ Style pour le main interne qui contient les articles */*

```
body.main-container main#main {  
    display: block;  
    /* Assurez-vous que les articles sont affichés en colonne */  
}
```

/ Style pour la section haute */*

```
.haut {  
    padding: 0.5em;  
    min-height: 200px;  
    margin-bottom: 1em;  
    width: 100%;  
}
```

/ Style pour les informations */*

```
.infos {  
    font-style: italic;  
}
```

/ Style pour les colonnes de widgets */*

```
#colonneDroite,  
#colonneGauche {  
    min-height: 200px;  
    padding: 0.5em;  
    margin-bottom: 1em;  
}
```

/ Style pour les div dans les colonnes de widgets */*

```
#colonneDroite div,  
#colonneGauche div {  
    margin-bottom: 3em;  
}
```



/ Style pour les listes dans les colonnes de widgets */*

```
#colonneDroite ul,  
#colonneGauche ul {  
    list-style-type: none;  
}
```

/=====*/*

/ Style pour la section basse */*

```
#bas {  
    padding: 0.5em;  
    margin-bottom: 1em;  
    display: flex;  
}
```

/ Style pour les widgets en bas */*

```
.widget-bas {  
    margin-left: 2em;  
    min-width: 200px;  
}
```

/ Style pour le pied de page */*

```
footer {  
    padding: 0.5em;  
}
```

/ Style pour le menu du pied de page */*

```
footer ul {  
    list-style-type: none;  
    margin-bottom: 0.5em;  
    text-align: center;  
    font-family: "Playwrite NO", cursive;  
    font-size: 0.7em;  
}
```

/ Style pour les éléments de liste dans le menu du pied de page */*

```
footer ul li {  
    display: inline-block;  
}
```

/ Style pour les éléments de liste dans le menu du pied de page, sauf le premier */*

```
footer ul li:not(:first-of-type) {
```



```
margin-left: 0.5em;
}

/* Style pour le copyright */
#copyright {
    text-align: center;
    font-family: "Playwrite NO", cursive;
    font-size: 0.7em;
}

/* Style pour les informations du pied de page */
#footer_info {
    display: flex;
    justify-content: space-between;
    font-family: "Playwrite NO", cursive;
    font-size: 0.7em;
}

/* Style pour la partie droite du pied de page */
#footer_right {
    display: flex;
    flex-direction: row-reverse;
}

/* Style pour les div dans la partie droite du pied de page */
#footer_right div {
    margin-left: 15px;
}

/* Style pour les icônes dans la partie droite du pied de page */
#footer_right i {
    font-size: 2.5em;
}

/* Style pour le bouton de basculement du menu */
.toggle-nav {
    display: none !important;
}

/* Styles pour le bouton de retour en haut de page */
/* Flèche pour le retour vers le haut de la page */
.fleche-hdpage i {
    /* Masqué par défaut */
}
```



```
/*display: none;*/
font-size: 1.875em;
position: fixed;
/* Position en bas */
bottom: 1.875em;
/* Position à droite */
right: 1.875em;
/* Couleur de fond */
background: #0a0a0a;
/* Couleur du texte */
color: white;
/* Padding pour augmenter la taille cliquable */
padding: 10px;
/* Bordures arrondies pour une forme circulaire */
border-radius: 50%;
/* Curseur en forme de pointeur */
cursor: pointer;
/* Assure que le bouton est au-dessus des autres éléments */
z-index: 1000;
opacity: 0.5;
transition: opacity 1s ease-in-out;
/* Ombre légère */
box-shadow: 0 2px 8px rgba(0, 0, 0, 0.2);

/* Empêche le bouton d'affecter la largeur totale de la page */
max-width: 100%;
box-sizing: border-box;
}

/* Style pour la carte */
#map {
  height: 400px;
  width: 100%;
  margin-bottom: 20px;
}

/* Styles pour la galerie d'art */
.art-gallery {
  display: flex;
  flex-wrap: wrap;
  gap: 20px;
}
```



/ Style pour les éléments de la galerie d'art */*

```
.art-gallery-item {  
    flex: 1 1 calc(33.333% - 20px);  
    box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);  
    transition: transform 0.3s ease;  
}
```

/ Style pour les éléments de la galerie d'art au survol */*

```
.art-gallery-item:hover {  
    transform: translateY(-10px) scale(1.05);  
    box-shadow: 0 10px 20px rgba(0, 0, 0, 0.2);  
}
```

/ Style pour les images dans les éléments de la galerie d'art */*

```
.art-gallery-item img {  
    width: 100%;  
    height: auto;  
}
```

/ Style pour les titres dans les éléments de la galerie d'art */*

```
.art-gallery-item h2 {  
    font-size: 1.5rem;  
    margin: 10px 0;  
    text-align: center;  
}
```

/ Style pour les paragraphes dans les éléments de la galerie d'art */*

```
.art-gallery-item p {  
    text-align: center;  
}
```

/ Style pour les filtres de la galerie d'art */*

```
.art-gallery-filters {  
    margin-bottom: 20px;  
}
```

/ Style pour le conteneur du bouton de chargement de plus de contenu */*

```
.load-more-container {  
    text-align: center;  
    margin-top: 20px;  
}
```

/ Style pour le bouton de chargement de plus de contenu */*



```
#load-more {
    padding: 10px 20px;
    font-size: 16px;
    cursor: pointer;
}

/*=====*/

/* Styles pour les écrans de largeur minimale de 480px et maximale de 768px */
@media screen and (min-width: 480px) and (max-width: 768px) {

    /* Style pour le corps de la page */
    body {
        max-width: 100%;
        /* Largeur maximale de 80% */
        margin: auto;
        /* Centré horizontalement */
    }

    /* Style pour les sections avec une barre latérale */
    .avecAside {
        width: 70%;
        /* Largeur de 70% */
    }

    /* Style pour les colonnes de widgets */
    #colonneDroite,
    #colonneGauche {
        width: 30%;
        /* Largeur de 30% */
    }

    /* Style pour le champ de recherche */
    #recherche {
        display: block;
        /* Affiché en bloc */
    }

    /* Style pour les éléments de liste dans le menu principal */
    nav#menuPrincipal ul li {
        display: inline-block;
        /* Affiché en ligne */
    }
}
```



```
}  
}  
/*=====
```

```
/* Styles pour les écrans de largeur maximale de 768px */  
@media screen and (max-width: 768px) {
```

```
/* Style pour le corps de la page */  
body {  
    max-width: 100%;  
    margin: auto;  
}
```

```
/* Style pour le main principal */  
body.main-container main {  
    display: block;  
    flex-direction: column;  
}
```

```
/* Style pour le main interne qui contient les articles */  
body.main-container main#main {  
    display: block;  
    width: 100%;  
}
```

```
/* Style pour les colonnes de widgets */  
#colonneDroite,  
#colonneGauche {  
    width: 100%;  
    margin-top: 20px;  
    /* Ajoutez un espace entre les articles et la sidebar */  
}
```

```
/* Style pour le pied de page */  
footer {  
    padding: 0.5em;  
    text-align: center;  
}
```

```
/* Style pour l'élément <aside id="bas"> */  
aside#bas {  
    display: flex;
```



```
flex-direction: column;
align-items: center;
width: 100%;
}
```

```
/* Style pour les widgets en bas */
```

```
.widget-bas {
margin-left: 0;
min-width: 100%;
display: block;
margin-bottom: 1em;
}
```

```
/* Style pour le menu du pied de page */
```

```
footer ul {
list-style-type: none;
margin-bottom: 0.5em;
text-align: center;
font-family: "Playwrite NO", cursive;
font-size: 0.7em;
}
```

```
/* Style pour les éléments de liste dans le menu du pied de page */
```

```
footer ul li {
display: block;
margin-bottom: 0.5em;
}
```

```
/* Style pour le copyright */
```

```
#copyright {
text-align: center;
font-family: "Playwrite NO", cursive;
font-size: 0.7em;
}
```

```
/* Style pour les informations du pied de page */
```

```
#footer_info {
display: flex;
flex-direction: column;
align-items: center;
font-family: "Playwrite NO", cursive;
font-size: 0.7em;
}
```



```
/* Style pour la partie droite du pied de page */
#footer_right {
    display: flex;
    flex-direction: column;
    align-items: center;
}

/* Style pour les div dans la partie droite du pied de page */
#footer_right div {
    margin-left: 0;
    margin-bottom: 0.5em;
}

/* Style pour les icônes dans la partie droite du pied de page */
#footer_right i {
    font-size: 2.5em;
}

/* Style pour l'affichage des produits WooCommerce */
.woocommerce-products-main {
    grid-template-columns: repeat(auto-fill, minmax(200px, 1fr));
    gap: 20px;
    margin-left: 5px;
}

/* Style pour le bouton de basculement du menu */
.toggle-nav {
    top: 10px;
    padding: 5px;
    margin: 5px;
    display: inline-block !important;
    color: #000000;
    color: #fff;
    transition: color linear 0.15s;
}

/* Style pour le bouton de basculement du menu au survol et actif */
.toggle-nav:hover,
.toggle-nav.active {
    text-decoration: none;
    color: #000000;
}
```



/ Style pour l'icône du menu mobile */*

```
#mobile-menu-icon {  
  display: block;  
}
```

/ Style pour le menu principal */*

```
#menuPrincipal {  
  display: none;  
}
```

/ Style pour le menu principal ouvert */*

```
#menuPrincipal.open {  
  display: block;  
  justify-content: right;  
  width: 50%;  
  position: absolute;  
  top: 70px;  
  margin-right: 0px;  
  margin-left: 30px;  
  background-color: #fff;  
  color: #000000;  
  z-index: 1000;  
  opacity: 0.9;  
  transition: opacity 1s ease-in-out;  
  box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);  
}
```

/ Style pour le menu */*

```
nav#menuPrincipal {  
  margin-bottom: 1em;  
  padding-left: 0.5em;  
}
```

/ Style pour les listes dans le menu */*

```
nav#menuPrincipal ul {  
  flex-direction: column;  
  list-style-type: none;  
  padding: 0;  
  display: none;  
}
```

/ Style pour les éléments de liste dans le menu */*



```
nav#menuPrincipal ul li {  
    display: block;  
    text-align: center;  
    padding: 10px 0;  
    margin-bottom: .5em;  
    position: relative;  
}
```

/ Style pour les liens dans les éléments de liste du menu */*

```
nav#menuPrincipal ul li a {  
    display: block;  
    text-decoration: none;  
    background-color: #fff;  
    color: #000000;  
    padding: 0.5em 1em;  
    border-radius: 10px;  
    text-align: center;  
    opacity: 0.9;  
    transition: opacity 1s ease-in-out;  
    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);  
}
```

/ Style pour les sous-menus */*

```
nav#menuPrincipal ul li ul li {  
    display: block;  
    width: 100%;  
    text-align: center;  
    border-top: 1px solid transparent;  
    padding: 0;  
    z-index: 1000;  
    opacity: 0.9;  
    transition: opacity 1s ease-in-out;  
    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);  
}
```

/ Style pour les liens dans les sous-menus */*

```
nav#menuPrincipal ul li ul li a {  
    padding: 0.5em 0;  
}
```

/ Style pour les sous-menus */*

```
nav#menuPrincipal ul li ul {  
    width: 100%;
```



```
position: absolute;
left: 0;
z-index: 1000;
display: none;
opacity: 0.9;
transition: opacity 1s ease-in-out;
box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);
}

/* Style pour les sous-menus au survol */
/* nav#menuPrincipal ul li: hover>ul {
    display: block;
} */

/* Par exemple, changer le comportement du sous-menu pour les écrans mobiles */
ul.sub-menu {
    display: none;
    position: static;
    background-color: #fff;
    z-index: 1000;
    opacity: 0.5;
    transition: opacity 1s ease-in-out;
    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);
}

/* Afficher les sous-menus lorsqu'ils sont activés via JavaScript */
.menu-item.active>ul.sub-menu {
    display: block;
}

/* Gestion de l'ouverture du sous-menu via un bouton sur mobile */
.menu-item>a {
    cursor: pointer;
}

}

/*=====*/
/* Styles pour les écrans de bureau (largeur minimale de 769px) */
@media screen and (min-width: 769px) {
    /* Style pour les éléments de liste dans le menu principal */
    nav#menuPrincipal ul li {
        display: inline-block;
        /* Affiché en ligne */
    }
}
}
```



functions.php :

```
<?php
// Définition des paramètres de l'image du bandeau
function mzlleFlo_custom_header_setup()
{
    $header_args = array(
        'flex-height' => true,
        // 'height' => 288,
        'flex-width' => true,
        //'width' => 560,
        'default-image' => get_template_directory_uri() . '/images/logo6_medium.gif',
    );
    add_theme_support('custom-header', $header_args);
    add_theme_support('post-thumbnails');
}
add_action('after_setup_theme', 'mzlleFlo_custom_header_setup');

// Activer l'option de couleur d'arrière-plan avec une couleur par défaut
add_theme_support('custom-background', array('default-color' => 'cccccc'));

// Fonction pour ajouter les styles CSS du thème
function mzlleFlo_scripts()
{
    // Enregistrer et enfiler le style principal du thème depuis le dossier css
    wp_enqueue_style('mzlleFlo-style', get_template_directory_uri() . '/style.css');

    // Enregistrer et enfiler le style principal de WooCommerce
    wp_enqueue_style('woocommerce-general', get_template_directory_uri() . '/woocommerce.css');
    // Utilise le style par défaut de WooCommerce
    wp_enqueue_style('woocommerce-general',
    plugins_url('woocommerce/assets/css/woocommerce.css'));

    // Enregistrer et enfiler le style FontAwesome
    wp_enqueue_style('font-awesome', 'https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.4/css/all.min.css');

    // Lightbox JS and CSS from CDN
    //wp_enqueue_script('lightbox-js',
'https://cdnjs.cloudflare.com/ajax/libs/lightbox2/2.11.3/js/lightbox.min.js', array('jquery'), null, true);
    //wp_enqueue_style('lightbox-css',
```




```
'https://cdnjs.cloudflare.com/ajax/libs/lightbox2/2.11.3/css/lightbox.min.css');
```

```
// Charger les styles spécifiques pour le modèle "Full Width"
if (is_page_template('full-width.php')) {
    wp_enqueue_style('full-width-style', get_template_directory_uri() . '/css/full-width.css');
}

// Enregistrer et enfiler le script personnalisé
wp_enqueue_script('custom-script', get_template_directory_uri() . '/js/scripts.js');
wp_enqueue_script('custom-menu-script', get_template_directory_uri() . '/js/custom-menu.js',
array('jquery'), null, true);

// Localiser le script avec la URL AJAX
wp_localize_script('custom-script', 'my_ajax_object', array(
    'ajax_url' => admin_url('admin-ajax.php')
));
}

// Ajouter les styles CSS avec le hook wp_enqueue_scripts
add_action('wp_enqueue_scripts', 'mzlleFlo_scripts');

function mzlleFlo_custom_admin_scripts($hook)
{
    if ($hook == 'widgets.php' || $hook == 'customize.php') {
        // Charger seulement les scripts nécessaires pour les widgets ou le customizer
        wp_enqueue_script('custom-widget-script', get_template_directory_uri() . '/js/custom-
widget.js');
    }
}
add_action('admin_enqueue_scripts', 'mzlleFlo_custom_admin_scripts');

// Enregistrer les emplacements des menus
if (function_exists('register_nav_menus')) {
    register_nav_menus(array(
        'principal' => 'Menu principal',
        'footer' => 'Menu footer',
    ));
}

// Activer le support pour les balises de titre dynamiques
add_theme_support('title-tag');

// Ajout des options personnalisées avec l'objet $wp_customize et les méthodes add_setting,
```



add_control, add_section, fonction pour personnaliser le thème dans le customizer de WordPress

function mzlleFlo_customize_register(\$wp_customize)

{

// Vérifier si selective_refresh est défini : ajoute des rafraîchissements partiels pour certains éléments

//Nous souhaitons ajouter l'icône de crayon au sein du "Live Preview" dans Apparence > Personnaliser :

// Nous devons utiliser pour cela selective_refresh->add_partial() pour chaque option à \$wp_customize .

//Nous ajoutons donc le code suivant à la fonction mzlleFlo_customize_register() :

if (isset(\$wp_customize->selective_refresh)) {

//En premier paramètre de add_partial(), nous spécifions le nom de l'option (existante ou personnalisée)

//et en second paramètre nous passons un tableau dans lequel nous définissons l'élément HTML devant lequel l'icône de crayon sera ajoutée.

// Ajouter des partiels pour le rafraîchissement sélectif

\$wp_customize->selective_refresh->add_partial('blogname', array('selector' => '.site-title'));

\$wp_customize->selective_refresh->add_partial('blogdescription', array('selector' => '.site-description'));

\$wp_customize->selective_refresh->add_partial('header-image', array('selector' => '#logo'));

\$wp_customize->selective_refresh->add_partial('mzlleFlo_credits', array('selector' => '#copyright'));

\$wp_customize->selective_refresh->add_partial('my_layout_width', array('selector' => '#colonneDroite'));

}

// Ajoute des paramètres et des contrôles pour personnaliser le thème

// couleur du texte

\$wp_customize->add_setting('text_color', array(

'default' => '#000000',

'transport' => 'refresh',

'sanitize_callback' => 'sanitize_hex_color',

));

\$wp_customize->add_control(new WP_Customize_Color_control(\$wp_customize, 'text_color', array(

'label' => 'Couleur du texte',

'section' => 'colors',

'settings' => 'text_color',

)));



// Ajouter une section pour le footer

```
$wp_customize->add_section('my_footer', array(
    'title' => 'Footer',
    'priority' => 120,
));
```

// Ajouter une section pour la mise en page "Layout"

```
$wp_customize->add_section('my_layout', array(
    'title' => 'Layout',
    'priority' => 60,
));
```

// Ajouter un paramètre et un contrôle pour la largeur de la sidebar

```
$wp_customize->add_setting('my_layout_width', array(
    'default' => 25, // Valeur par défaut: 25%
    'capability' => 'edit_theme_options',
    'sanitize_callback' => 'esc_attr',
));
```

// Ajout du contrôleur pour "layout_width"

```
$wp_customize->add_control('my_layout_width', array(
    'settings' => 'my_layout_width',
    'label' => "Largeur de la sidebar",
    'description' => '<em>20% à 40%</em>', // Description en italique
    'section' => 'my_layout',
    'type' => 'range',
    'input_attrs' => array(
        'min' => 20,
        'max' => 40,
        'step' => 5,
    )
));
```

// Ajouter un paramètre et un contrôle pour afficher le copyright, seuls les users pouvant éditer les options pourront y accéder

```
$wp_customize->add_setting('mzlleFlo_credits', array(
    'default' => 'oui',
    'capability' => 'edit_theme_options',
    'sanitize_callback' => 'esc_attr',
));
```

//Ajout du contrôleur 'mzlleFlo_credit

```
$wp_customize->add_control('mzlleFlo_credits', array(
    'settings' => 'mzlleFlo_credits',
    'label' => "Souhaitez-vous montrer le copyright ?",
));
```



```
'section' => 'my_footer',  
'type' => 'radio',  
'choices' => array(  
    'oui' => "Oui",  
    'non' => "Non",  
)  
));
```

// Ajouter des paramètres et des contrôles pour les informations de contact dans le footer

// Ajout de l'option "Téléphone"

```
$wp_customize->add_setting('mzlleFlo_tel', array(  
    'default' => "",  
    'capability' => 'edit_theme_options',  
    'sanitize_callback' => 'esc_attr',  
));  
$wp_customize->add_control('mzlleFlo_tel', array(  
    'settings' => 'mzlleFlo_tel',  
    'label' => "Téléphone",  
    'section' => 'my_footer',  
    'type' => 'input',  
    'input_attrs' => array('placeholder' => '0123456789'),  
));
```

// Ajout de l'option "Adresse"

```
$wp_customize->add_setting('mzlleFlo_adresse', array(  
    'default' => "",  
    'capability' => 'edit_theme_options',  
    'sanitize_callback' => 'esc_attr',  
));  
$wp_customize->add_control('mzlleFlo_adresse', array(  
    'settings' => 'mzlleFlo_adresse',  
    'label' => "Adresse",  
    'section' => 'my_footer',  
    'type' => 'textarea',  
    'input_attrs' => array('placeholder' => 'votre adresse'),  
));
```

// Ajout de l'option "URL page Facebook"

```
$wp_customize->add_setting('mzlleFlo_facebook', array(  
    'default' => "",  
    'capability' => 'edit_theme_options',  
    'sanitize_callback' => 'sanitize_textarea_field',  
));
```



```
$wp_customize->add_control('mzlleFlo_facebook', array(
    'settings' => 'mzlleFlo_facebook',
    'label' => "URL page Facebook",
    'section' => 'my_footer',
    'type' => 'url',
    'input_attrs' => array('placeholder' => 'http://www.facebook.fr'),
));
```

// Ajout de l'option "URL page Tumblr"

```
$wp_customize->add_setting('mzlleFlo_tumblr', array(
    'default' => "",
    'capability' => 'edit_theme_options',
    'sanitize_callback' => 'esc_attr',
));
$wp_customize->add_control('mzlleFlo_tumblr', array(
    'settings' => 'mzlleFlo_tumblr',
    'label' => "URL page Tumblr",
    'section' => 'my_footer',
    'type' => 'url',
    'input_attrs' => array('placeholder' => 'https://'),
));
```

// Section pour Google Maps

```
$wp_customize->add_section('my_google', array(
    'title' => 'Page contact - Google map',
    'priority' => 50,
));
$wp_customize->add_setting('my_google_map', array(
    'capability' => 'edit_theme_options',
    'sanitize_callback' => 'esc_attr',
    'default' => false,
));
$wp_customize->add_control('my_google_map', array(
    'settings' => 'my_google_map',
    'label' => "Afficher la Google map ?",
    'description' => "(page contact - juste avant le footer)",
    'section' => 'my_google',
    'type' => 'checkbox',
));
```

```
$wp_customize->add_setting('my_google_api', array(
    'capability' => 'edit_theme_options',
    'sanitize_callback' => 'esc_attr',
```



```
));  
$wp_customize->add_control('my_google_api', array(  
    'settings' => 'my_google_api',  
    'label' => "votre clé API",  
    'section' => 'my_google',  
    'description' => 'https://developers.google.com/maps/documentation/embed/embedding-map',  
    'type' => 'url',  
));  
$wp_customize->add_setting('my_google_lat', array(  
    'capability' => 'edit_theme_options',  
    'sanitize_callback' => 'esc_attr',  
));  
$wp_customize->add_control('my_google_lat', array(  
    'settings' => 'my_google_lat',  
    'label' => "Latitude",  
    'section' => 'my_google',  
    'type' => 'input',  
));  
$wp_customize->add_setting('my_google_long', array(  
    'capability' => 'edit_theme_options',  
    'sanitize_callback' => 'esc_attr',  
));  
$wp_customize->add_control('my_google_long', array(  
    'settings' => 'my_google_long',  
    'label' => "Longitude",  
    'section' => 'my_google',  
    'type' => 'input',  
));  
$wp_customize->add_setting('my_google_zoom', array(  
    'default' => 10,  
    'capability' => 'edit_theme_options',  
    'sanitize_callback' => 'esc_attr',  
));  
$wp_customize->add_control('my_google_zoom', array(  
    'settings' => 'my_google_zoom',  
    'label' => "Zoom",  
    'section' => 'my_google',  
    'type' => 'range',  
    'input_attrs' => array(  
        'min' => 0,  
        'max' => 21,  
        'step' => 1,  
    )  
)
```



```
));
```

// Section pour les polices

```
$wp_customize->add_section('typography', array(  
    'title' => 'Typographie',  
    'priority' => 30,  
));
```

// Polices Google

```
$google_fonts = array(  
    'default' => 'Police par défaut',  
    'Roboto' => 'Roboto',  
    'Open Sans' => 'Open Sans',  
    'Lato' => 'Lato',  
    'Montserrat' => 'Montserrat',  
    'Great Vibes' => 'Great Vibes',  
    'Send Flowers' => 'Send Flowers',  
    'Playwrite NO' => 'Playwrite NO',  
);
```

// Menu

```
$wp_customize->add_setting('menu_font_family', array(  
    'default' => 'Playwrite NO',  
    'sanitize_callback' => 'sanitize_text_field',  
));  
$wp_customize->add_control('menu_font_family', array(  
    'label' => 'Police pour le menu',  
    'section' => 'typography',  
    'type' => 'select',  
    'choices' => $google_fonts,  
));
```

```
$wp_customize->add_setting(  
    'menu_font_size',  
    array(  
        'default' => 16,  
        'sanitize_callback' => 'absint',  
    )  
);  
$wp_customize->add_control('menu_font_size', array(  
    'label' => 'Taille de la police pour le menu',  
    'section' => 'typography',  
    'type' => 'number',
```



```
'input_attrs' => array(
    'min' => 10,
    'max' => 100,
    'step' => 1,
),
));

$wp_customize->add_setting('menu_font_bold', array(
    'default' => false,
    'sanitize_callback' => 'mzlleFlo_sanitize_checkbox',
));
$wp_customize->add_control('menu_font_bold', array(
    'label' => 'Gras',
    'section' => 'typography',
    'type' => 'checkbox',
));

$wp_customize->add_setting('menu_font_italic', array(
    'default' => false,
    'sanitize_callback' => 'mzlleFlo_sanitize_checkbox',
));
$wp_customize->add_control('menu_font_italic', array(
    'label' => 'Italique',
    'section' => 'typography',
    'type' => 'checkbox',
));

// Entête
$wp_customize->add_setting('header_font_family', array(
    'default' => 'Roboto',
    'sanitize_callback' => 'sanitize_text_field',
));
$wp_customize->add_control('header_font_family', array(
    'label' => 'Police pour l\'entête',
    'section' => 'typography',
    'type' => 'select',
    'choices' => $google_fonts,
));

$wp_customize->add_setting('header_font_size', array(
    'default' => 24,
    'sanitize_callback' => 'absint',
));
```




```
$wp_customize->add_control('header_font_size', array(
    'label' => 'Taille de la police pour l\'entête',
    'section' => 'typography',
    'type' => 'number',
    'input_attrs' => array(
        'min' => 10,
        'max' => 100,
        'step' => 1,
    ),
));
```

```
$wp_customize->add_setting('header_font_bold', array(
    'default' => false,
    'sanitize_callback' => 'mzlleFlo_sanitize_checkbox',
));
$wp_customize->add_control('header_font_bold', array(
    'label' => 'Gras',
    'section' => 'typography',
    'type' => 'checkbox',
));
```

```
$wp_customize->add_setting('header_font_italic', array(
    'default' => false,
    'sanitize_callback' => 'mzlleFlo_sanitize_checkbox',
));
$wp_customize->add_control('header_font_italic', array(
    'label' => 'Italique',
    'section' => 'typography',
    'type' => 'checkbox',
));
```

// Texte global

```
$wp_customize->add_setting('body_font_family', array(
    'default' => 'Open Sans',
    'sanitize_callback' => 'sanitize_text_field',
));
$wp_customize->add_control('body_font_family', array(
    'label' => 'Police pour le texte global',
    'section' => 'typography',
    'type' => 'select',
    'choices' => $google_fonts,
));
```



```
$wp_customize->add_setting('body_font_size', array(
    'default' => 16,
    'sanitize_callback' => 'absint',
));
```

```
$wp_customize->add_control('body_font_size', array(
    'label' => 'Taille de la police pour le texte global',
    'section' => 'typography',
    'type' => 'number',
    'input_attrs' => array(
        'min' => 10,
        'max' => 100,
        'step' => 1,
    ),
));
```

```
$wp_customize->add_setting('body_font_bold', array(
    'default' => false,
    'sanitize_callback' => 'mzlleFlo_sanitize_checkbox',
));
```

```
$wp_customize->add_control('body_font_bold', array(
    'label' => 'Gras',
    'section' => 'typography',
    'type' => 'checkbox',
));
```

```
$wp_customize->add_setting('body_font_italic', array(
    'default' => false,
    'sanitize_callback' => 'mzlleFlo_sanitize_checkbox',
));
```

```
$wp_customize->add_control('body_font_italic', array(
    'label' => 'Italique',
    'section' => 'typography',
    'type' => 'checkbox',
));
```

// Catégories

```
$wp_customize->add_setting('categories_font_family', array(
    'default' => 'Open Sans',
    'sanitize_callback' => 'sanitize_text_field',
));
```

```
$wp_customize->add_control('categories_font_family', array(
    'label' => 'Police pour les catégories',
    'section' => 'typography',
));
```



```
'type' => 'select',
'choices' => $google_fonts,
));

$wp_customize->add_setting('categories_font_size', array(
    'default' => 24,
    'sanitize_callback' => 'absint',
));
$wp_customize->add_control('categories_font_size', array(
    'label' => 'Taille de la police pour les catégories',
    'section' => 'typography',
    'type' => 'number',
    'input_attrs' => array(
        'min' => 10,
        'max' => 100,
        'step' => 1,
    ),
));

$wp_customize->add_setting('categories_font_bold', array(
    'default' => false,
    'sanitize_callback' => 'mzlleFlo_sanitize_checkbox',
));
$wp_customize->add_control('categories_font_bold', array(
    'label' => 'Gras',
    'section' => 'typography',
    'type' => 'checkbox',
));

$wp_customize->add_setting('categories_font_italic', array(
    'default' => false,
    'sanitize_callback' => 'mzlleFlo_sanitize_checkbox',
));
$wp_customize->add_control('categories_font_italic', array(
    'label' => 'Italique',
    'section' => 'typography',
    'type' => 'checkbox',
));

// Texte footer
$wp_customize->add_setting('footer_font_family', array(
    'default' => 'Open Sans',
    'sanitize_callback' => 'sanitize_text_field',
```



```
));  
$wp_customize->add_control('footer_font_family', array(  
    'label' => 'Police pour le footer',  
    'section' => 'typography',  
    'type' => 'select',  
    'choices' => $google_fonts,  
));  
  
$wp_customize->add_setting('footer_font_size', array(  
    'default' => 16,  
    'sanitize_callback' => 'absint',  
));  
$wp_customize->add_control('footer_font_size', array(  
    'label' => 'Taille de la police pour le footer',  
    'section' => 'typography',  
    'type' => 'number',  
    'input_attrs' => array(  
        'min' => 10,  
        'max' => 100,  
        'step' => 1,  
    ),  
));  
  
$wp_customize->add_setting('footer_font_bold', array(  
    'default' => false,  
    'sanitize_callback' => 'mzlleFlo_sanitize_checkbox',  
));  
$wp_customize->add_control('footer_font_bold', array(  
    'label' => 'Gras',  
    'section' => 'typography',  
    'type' => 'checkbox',  
));  
  
$wp_customize->add_setting('footer_font_italic', array(  
    'default' => false,  
    'sanitize_callback' => 'mzlleFlo_sanitize_checkbox',  
));  
$wp_customize->add_control('footer_font_italic', array(  
    'label' => 'Italique',  
    'section' => 'typography',  
    'type' => 'checkbox',  
));
```



// Ajouter une section pour la galerie d'art

```
$wp_customize->add_section('art_gallery_section', array(
    'title' => __('Art Gallery', 'textdomain'),
    'priority' => 30,
));
```

// Ajouter un contrôle pour le nombre d'images par page

```
$wp_customize->add_setting('art_gallery_images_per_page', array(
    'default' => 10,
    'sanitize_callback' => 'absint',
));
```

```
$wp_customize->add_control(
    'art_gallery_images_per_page',
    array(
        'label' => __('Nombre d\'images par page', 'textdomain'),
        'section' => 'art_gallery_section',
        'type' => 'number',
    )
);
```

// Ajouter un contrôle pour la taille des images

```
$wp_customize->add_setting('art_gallery_image_size', array(
    'default' => 'medium',
    'sanitize_callback' => 'sanitize_text_field',
));
```

```
$wp_customize->add_control('art_gallery_image_size', array(
    'label' => __('Taille des images', 'textdomain'),
    'section' => 'art_gallery_section',
    'type' => 'select',
    'choices' => array(
        'thumbnail' => __('Petite', 'textdomain'),
        'medium' => __('Moyenne', 'textdomain'),
        'large' => __('Grande', 'textdomain'),
    ),
));
```

// Ajouter un contrôle pour l'effet de transition

```
$wp_customize->add_setting('art_gallery_transition_effect', array(
    'default' => 'fade',
    'sanitize_callback' => 'sanitize_text_field',
));
```



```
$wp_customize->add_control('art_gallery_transition_effect', array(
    'label' => __('Effet de transition', 'textdomain'),
    'section' => 'art_gallery_section',
    'type' => 'select',
    'choices' => array(
        'fade' => __('Fade', 'textdomain'),
        'slide' => __('Slide', 'textdomain'),
    ),
));
```

// Ajouter un contrôle pour la couleur de fond de la lightbox

```
$wp_customize->add_setting('art_gallery_lightbox_bg_color', array(
    'default' => '#000000',
    'sanitize_callback' => 'sanitize_hex_color',
));
```

```
$wp_customize->add_control(new WP_Customize_Color_Control($wp_customize,
'art_gallery_lightbox_bg_color', array(
    'label' => __('Couleur de fond de la lightbox', 'textdomain'),
    'section' => 'art_gallery_section',
)));
```

// Ajouter un contrôle pour la couleur de la bordure des images

```
$wp_customize->add_setting('art_gallery_image_border_color', array(
    'default' => '#ffffff',
    'sanitize_callback' => 'sanitize_hex_color',
));
```

```
$wp_customize->add_control(new WP_Customize_Color_Control($wp_customize,
'art_gallery_image_border_color', array(
    'label' => __('Couleur de la bordure des images', 'textdomain'),
    'section' => 'art_gallery_section',
)));
```

// Ajouter un contrôle pour l'affichage des légendes

```
$wp_customize->add_setting('art_gallery_show_captions', array(
    'default' => true,
    'sanitize_callback' => 'sanitize_text_field',
));
```

```
$wp_customize->add_control('art_gallery_show_captions', array(
    'label' => __('Afficher les légendes', 'textdomain'),
```



```
'section' => 'art_gallery_section',  
'type' => 'checkbox',  
));
```

// Ajouter un contrôle pour l'affichage des numéros d'image

```
$wp_customize->add_setting('art_gallery_show_image_numbers', array(  
    'default' => true,  
    'sanitize_callback' => 'sanitize_text_field',  
));
```

```
$wp_customize->add_control('art_gallery_show_image_numbers', array(  
    'label' => __('Afficher les numéros d\'image', 'textdomain'),  
    'section' => 'art_gallery_section',  
    'type' => 'checkbox',  
));
```

// Ajouter un contrôle pour la durée de l'animation

```
$wp_customize->add_setting('art_gallery_animation_duration', array(  
    'default' => 600,  
    'sanitize_callback' => 'absint',  
));
```

```
$wp_customize->add_control('art_gallery_animation_duration', array(  
    'label' => __('Durée de l\'animation (ms)', 'textdomain'),  
    'section' => 'art_gallery_section',  
    'type' => 'number',  
));
```

// Ajouter un contrôle pour l'affichage des flèches de navigation

```
$wp_customize->add_setting('art_gallery_show_navigation_arrows', array(  
    'default' => true,  
    'sanitize_callback' => 'sanitize_text_field',  
));
```

```
$wp_customize->add_control('art_gallery_show_navigation_arrows', array(  
    'label' => __('Afficher les flèches de navigation', 'textdomain'),  
    'section' => 'art_gallery_section',  
    'type' => 'checkbox',  
));
```

// Ajouter un contrôle pour l'affichage du bouton de fermeture

```
$wp_customize->add_setting('art_gallery_show_close_button', array(  
    'default' => true,
```



```
'sanitize_callback' => 'sanitize_text_field',
));

$wp_customize->add_control('art_gallery_show_close_button', array(
    'label' => __('Afficher le bouton de fermeture', 'textdomain'),
    'section' => 'art_gallery_section',
    'type' => 'checkbox',
));

// Enregistrer les nouvelles polices Google Fonts
function mzlleFlo_enqueue_google_fonts()
{
    $header_font = get_theme_mod('header_font_family', 'Roboto');
    $body_font = get_theme_mod('body_font_family', 'Open Sans');
    $categories_font = get_theme_mod('categories_font_family', 'Open Sans');
    $footer_font = get_theme_mod('footer_font_family', 'Open Sans');
    $menu_font = get_theme_mod('menu_font_family', 'Playwrite NO');

    $fonts = array_unique(array($header_font, $body_font, $categories_font, $footer_font,
    $menu_font));
    $font_query = implode('|', array_map('urlencode', $fonts));

    wp_enqueue_style('mzlleFlo-google-fonts', 'https://fonts.googleapis.com/css?family=' .
    $font_query, false);
}
add_action('wp_enqueue_scripts', 'mzlleFlo_enqueue_google_fonts');

function mzlleFlo_sanitize_checkbox($checked)
{
    return ((isset($checked) && true == $checked) ? true : false);
}

//Les fonctions seront exécutées lors de l'enregistrement des options par WordPress (hook
'customize_register').
// Ajouter l'action customize_register pour enregistrer les personnalisations de thème
add_action('customize_register', 'mzlleFlo_customize_register');

// Ajouter du CSS pour les personnalisations de thème

//Appliquer l'option de couleur du texte que nous avons paramétrée mais aussi la couleur du texte
d'entête que nous avons activée
//(la couleur d'arrière-plan s'applique déjà au body ), celle-ci sera appliquée au titre principal.
```




// Il est impossible d'insérer du php au sein du css, la solution est donc de gérer ceci dans le html balise <style>

// Afin d'associer les options de couleurs des éléments html souhaités, nous greffons son exécution lors de la construction du <head>

function theme_customize_css()

```
{
    $menu_font = get_theme_mod('menu_font_family', 'Playwrite NO');
    $menu_font_size = get_theme_mod('menu_font_size', 16);
    $menu_font_bold = get_theme_mod('menu_font_bold') ? 'bold' : 'normal';
    $menu_font_italic = get_theme_mod('menu_font_italic') ? 'italic' : 'normal';

    $header_font = get_theme_mod('header_font_family', 'Roboto');
    $header_font_size = get_theme_mod('header_font_size', 24);
    $header_font_bold = get_theme_mod('header_font_bold') ? 'bold' : 'normal';
    $header_font_italic = get_theme_mod('header_font_italic') ? 'italic' : 'normal';

    $body_font = get_theme_mod('body_font_family', 'Open Sans');
    $body_font_size = get_theme_mod('body_font_size', 16);
    $body_font_bold = get_theme_mod('body_font_bold') ? 'bold' : 'normal';
    $body_font_italic = get_theme_mod('body_font_italic') ? 'italic' : 'normal';

    $categories_font = get_theme_mod('categories_font_family', 'Open Sans');
    $categories_font_size = get_theme_mod('categories_font_size', 24);
    $categories_font_bold = get_theme_mod('categories_font_bold') ? 'bold' : 'normal';
    $categories_font_italic = get_theme_mod('categories_font_italic') ? 'italic' : 'normal';

    $footer_font = get_theme_mod('footer_font_family', 'Open Sans');
    $footer_font_size = get_theme_mod('footer_font_size', 24);
    $footer_font_bold = get_theme_mod('footer_font_bold') ? 'bold' : 'normal';
    $footer_font_italic = get_theme_mod('footer_font_italic') ? 'italic' : 'normal';

    // Art Gallery Settings
    $art_gallery_images_per_page = get_theme_mod('art_gallery_images_per_page', 10);
    $art_gallery_image_size = get_theme_mod('art_gallery_image_size', 'medium');
    $art_gallery_transition_effect = get_theme_mod('art_gallery_transition_effect', 'fade');
    $art_gallery_lightbox_bg_color = get_theme_mod('art_gallery_lightbox_bg_color', '#000000');
    $art_gallery_image_border_color = get_theme_mod('art_gallery_image_border_color', '#ffffff');
    $art_gallery_show_captions = get_theme_mod('art_gallery_show_captions', true);
    $art_gallery_show_image_numbers = get_theme_mod('art_gallery_show_image_numbers', true);
    $art_gallery_animation_duration = get_theme_mod('art_gallery_animation_duration', 600);
    $art_gallery_show_navigation_arrows = get_theme_mod('art_gallery_show_navigation_arrows',
true);
```



```
$art_gallery_show_close_button = get_theme_mod('art_gallery_show_close_button', true);
?>
<style type="text/css">
    nav#menuPrincipal ul li a {
        font-family: '<?php echo esc_attr($menu_font); ?>', sans-serif;
        font-size: <?php echo intval($menu_font_size); ?>px;
        font-weight: <?php echo esc_attr($menu_font_bold); ?>;
        font-style: <?php echo esc_attr($menu_font_italic); ?>;
    }

    body {
        font-family: '<?php echo esc_attr($body_font); ?>', sans-serif;
        font-size: <?php echo intval($body_font_size); ?>px;
        font-weight: <?php echo esc_attr($body_font_bold); ?>;
        font-style: <?php echo esc_attr($body_font_italic); ?>;
    }

    header h1 {
        font-family: '<?php echo esc_attr($header_font); ?>', sans-serif;
        font-size: <?php echo intval($header_font_size); ?>px;
        font-weight: <?php echo esc_attr($header_font_bold); ?>;
        font-style: <?php echo esc_attr($header_font_italic); ?>;
    }

    .category {
        font-family: '<?php echo esc_attr($categories_font); ?>', sans-serif;
        font-size: <?php echo intval($categories_font_size); ?>px;
        font-weight: <?php echo esc_attr($categories_font_bold); ?>;
        font-style: <?php echo esc_attr($categories_font_italic); ?>;
    }

    footer {
        font-family: '<?php echo esc_attr($footer_font); ?>', sans-serif;
        font-size: <?php echo intval($footer_font_size); ?>px;
        font-weight: <?php echo esc_attr($footer_font_bold); ?>;
        font-style: <?php echo esc_attr($footer_font_italic); ?>;
    }

    /* Art Gallery Styles */
    .art-gallery {
        /* Add your custom styles for the art gallery here */
    }
}
```



```
.art-gallery-item {
    /* Add your custom styles for the art gallery items here */
}

.lb-outerContainer {
    background-color: <?php echo esc_attr($art_gallery_lightbox_bg_color); ?>;
}

.lb-image {
    border-color: <?php echo esc_attr($art_gallery_image_border_color); ?>;
}

.lb-caption {
    display: <?php echo $art_gallery_show_captions ? 'block' : 'none'; ?>;
}

.lb-number {
    display: <?php echo $art_gallery_show_image_numbers ? 'block' : 'none'; ?>;
}

.lb-nav {
    display: <?php echo $art_gallery_show_navigation_arrows ? 'block' : 'none'; ?>;
}

.lb-close {
    display: <?php echo $art_gallery_show_close_button ? 'block' : 'none'; ?>;
}
</style>
<?php
}
// Ajoute l'action pour insérer le CSS personnalisé dans le head
add_action('wp_head', 'theme_customize_css');

// Enregistrer les sidebars via la méthode register_sidebar() au sein d'une fonction qui sera exécutée
lors de l'initialisation des widgets

// Fonction pour enregistrer les sidebars (widgets areas)
function mzlleFlo_register_sidebars()
{

    //Ce test permet à cette fonction d'être réécrite éventuellement par un thème enfant ou un plugin.
    if (function_exists('register_sidebar')) {
        // Enregistrer la sidebar principale
```



```
register_sidebar(array(
    'id' => 'sidebar-droite',
    'name' => 'Barre latérale principale',
    'description' => "Colonne de widgets qui apparait à droite",
    'before_widget' => "",
    'after_widget' => "",
    'before_title' => '<h3>',
    'after_title' => '</h3>',
));
register_sidebar(array(
    'id' => 'sidebar-bas',
    'name' => 'Zone de dépôt en pied de page',
    'description' => "Emplacement de widgets au-dessus du footer",
    'before_widget' => '<div class="widget-bas">',
    'after_widget' => '</div>',
    'before_title' => '<h3>',
    'after_title' => '</h3>',
));
register_sidebar(array(
    'id' => 'sidebar-contact',
    'name' => 'Barre latérale Contact',
    'description' => "Colonne de widgets de la page Contact qui apparait à gauche",
    'before_widget' => "",
    'after_widget' => "",
    'before_title' => '<h3>',
    'after_title' => '</h3>',
));
}
}
```

```
// Ajoute l'action pour enregistrer les sidebars
add_action('widgets_init', 'mzlleFlo_register_sidebars');
```

// Fonction ajouter des classes au <body> pour les articles et les catégories

```
function add_custom_body_classes($classes)
{
    if (is_single() || is_page() || is_category()) {
        $classes[] = 'main-container';
    }
    return $classes;
}
add_filter('body_class', 'add_custom_body_classes');
```



// Fonction pour modifier la longueur des extraits

function nbMotsExtraits(\$length)

{

 return 25;

}

add_filter('excerpt_length', 'nbMotsExtraits');

// Fonction pour modifier le texte "Lire la suite" des extraits

function finExtrait(\$more)

{

 return ' [→]';

}

add_filter('excerpt_more', 'finExtrait');

// Définir la largeur du contenu

if (!isset(\$content_width)) \$content_width = 900;

// Ajouter le support pour les commentaires sur les articles

if (is_singular()) wp_enqueue_script("comment-reply");

// Ajouter le support pour les liens de flux automatiques

add_theme_support('automatic-feed-links');

// Fonction pour ajouter les styles de l'éditeur

function mzlleFlo_add_editor_styles()

{

 add_editor_style(get_stylesheet_uri());

}

add_action('init', 'mzlleFlo_add_editor_styles');

// Modifier l'URL du login

function custom_login_url()

{

 return home_url('/welcome-mzlleflo');

}

add_filter('login_headerurl', 'custom_login_url');

//Bloquer l'authentification au back-office pendant 15 minutes après trois tentatives

function limit_login_attempts()

{

 \$max_attempts = 3;

 \$lockout_duration = 15 * 60; // 15 minutes



```
$attempts = get_transient('login_attempts_' . $_SERVER['REMOTE_ADDR']);

if ($attempts === false) {
    set_transient('login_attempts_' . $_SERVER['REMOTE_ADDR'], 1, $lockout_duration);
} elseif ($attempts >= $max_attempts) {
    $error = 'Vous avez dépassé le nombre maximum de tentatives de connexion. Réessayez dans 15 minutes.';
    wp_die($error, 'Erreur de connexion');
} else {
    set_transient('login_attempts_' . $_SERVER['REMOTE_ADDR'], $attempts + 1, $lockout_duration);
}
}
add_action('wp_login_failed', 'limit_login_attempts');
```

// Activer le support pour WooCommerce

```
function yourtheme_setup()
{
    add_theme_support('woocommerce');
}

add_action('after_setup_theme', 'yourtheme_setup');
```

// Wrappers pour le contenu WooCommerce

```
function yourtheme_woocommerce_wrapper_start()
{
    echo '<div id="primary" class="content-area">';
    echo '<main id="main" class="site-main">';
}

function yourtheme_woocommerce_wrapper_end()
{
    echo '</main></div>';
}

add_action('woocommerce_before_main_content', 'yourtheme_woocommerce_wrapper_start', 10);
add_action('woocommerce_after_main_content', 'yourtheme_woocommerce_wrapper_end', 10);
```

//RGPD : Ajouter la lecture des conditions avant validation de la commande.

```
add_action('woocommerce_review_order_before_submit', 'wpm_woocommerce_rgpd', 10);
function wpm_woocommerce_rgpd()
{
    ?>
    <p class="form-row terms">
        <input type="checkbox" class="input-checkbox" name="rgpd" id="rgpd">
```



```
<label for="rgpd" class="checkbox">J'ai lu et j'accepte la <a href="#">politique de confidentialité
du site*</a></strong></label>

</p>
<?php
}
```

// Erreur affichée si l'utilisateur ne coche pas la case

```
add_action('woocommerce_checkout_process', 'wpm_woocommerce_rgpd_erreur');
function wpm_woocommerce_rgpd_erreur()
{
    if (!(int) isset($_POST['rgpd'])) {
        wc_add_notice(__('<strong>Vous devez accepter la <a href="/politique-de-
confidentialite/">politique de confidentialité du site</a></strong>'), 'error');
    }
}
```

function desactiver_wp_editor_si_woocommerce()

```
{
    if (class_exists('WooCommerce')) {
        wp_deregister_script('wp-editor');
    }
}
add_action('wp_print_scripts', 'desactiver_wp_editor_si_woocommerce', 100);
```

// Disable Widgets Block Editor

```
add_filter('use_widgets_block_editor', '__return_false');
```

// Removing front end admin bar because it's ugly

```
add_filter('show_admin_bar', '__return_false');
```

//Script pour gérer les passerelles de paiement conditionnelles

```
add_filter('woocommerce_available_payment_gateways',
'manage_payment_gateways_based_on_transaction_type', 10, 1);
```

```
/**
```

**** Fonction pour gérer les passerelles de paiement en fonction du type de transaction.***

**** @param array \$available_gateways Les passerelles de paiement disponibles.***

**** @return array Les passerelles de paiement filtrées.***

****/***

```
function manage_payment_gateways_based_on_transaction_type($available_gateways)
{
```

// Ne pas appliquer les modifications dans l'administration



```
if (is_admin()) return $available_gateways;

// Initialiser le type de transaction
$type_of_transaction = "";

// Parcourir les articles du panier
foreach (WC()->cart->get_cart() as $cart_item_key => $cart_item) {
    $product = $cart_item['data'];
    if ($product->is_type('variation')) {
        // Récupérer l'attribut "Type de transaction"
        $type_of_transaction = $product->get_attribute('pa_type_de_transaction'); // Assurez-vous
        que le slug de l'attribut est correct
        error_log('Type de transaction: ' . $type_of_transaction); // Message de débogage
        break;
    }
}

// Si le type de transaction est "Vente", désactiver le prélèvement automatique
if ($type_of_transaction == 'Vente') {
    unset($available_gateways['stripe_sepa']);
}

// Si le type de transaction est "Location (mensuel)", désactiver la carte bancaire
elseif ($type_of_transaction == 'Location (mensuel)') {
    unset($available_gateways['stripe']);
}

// Retourner les passerelles de paiement filtrées
return $available_gateways;
}

//Script pour gérer les dépôts
add_action('woocommerce_before_calculate_totals', 'adjust_price_for_deposit', 10, 1);

/**
 * Fonction pour ajuster le prix total en fonction du type de transaction.
 *
 * @param WC_Cart $cart_object L'objet panier WooCommerce.
 */
function adjust_price_for_deposit($cart_object)
{
    // Parcourir les articles du panier
    foreach ($cart_object->get_cart() as $cart_item_key => $cart_item) {
```




```
$product = $cart_item['data'];
if ($product->is_type('variation')) {
    // Récupérer l'attribut "Type de transaction"
    $type_of_transaction = $product->get_attribute('pa_type_de_transaction'); // Assurez-vous
    que le slug de l'attribut est correct

    // Ajouter un message de débogage
    error_log('Type de transaction: ' . $type_of_transaction);

    // Si le type de transaction est "Location (mensuel)", ajuster le prix pour inclure la caution
    if ($type_of_transaction == 'Location (mensuel)') {
        $original_price = $product->get_price();
        $deposit = $original_price * 0.30; // 30% du prix de vente
        $cart_item['data']->set_price($original_price + $deposit);
    }
}
}
```

//Afficher la caution dans le panier : Ajouter un message personnalisé dans le panier

```
add_action('woocommerce_before_cart', 'display_deposit_message_in_cart');
```

/**

**** Fonction pour afficher un message de caution dans le panier.***

****/***

```
function display_deposit_message_in_cart()
```

```
{
```

```
    foreach (WC()->cart->get_cart() as $cart_item_key => $cart_item) {
```

```
        $product = $cart_item['data'];
```

```
        if ($product->is_type('variation')) {
```

```
            $type_of_transaction = $product->get_attribute('pa_type_de_transaction'); // Assurez-vous
```

```
            que le slug de l'attribut est correct
```

```
            // Ajouter un message de débogage
```

```
            error_log('Type de transaction: ' . $type_of_transaction);
```

```
            if ($type_of_transaction == 'Location (mensuel)') {
```

```
                $original_price = $product->get_price() / 1.30; // Calculer le prix original
```

```
                $deposit = $original_price * 0.30; // 30% du prix de vente
```

```
                echo '<p>Une caution de ' . wc_price($deposit) . ' (30% du prix de vente) sera prélevée pour
cette location.</p>';
```

```
            }
```



```

    }
}
}

```

//Afficher les informations de paiement sur les pages de produit : Ajouter un message personnalisé sur les pages de produit

```
add_action('woocommerce_single_product_summary', 'display_payment_info_on_product_page', 25);
```

```
/**
```

```
 * Fonction pour afficher les informations de paiement sur les pages de produit.
```

```
 */
```

```
function display_payment_info_on_product_page()
```

```
{
```

```
    global $product;
```

```
    if ($product->is_type('variation')) {
```

```
        $type_of_transaction = $product->get_attribute('pa_type_de_transaction'); // Assurez-vous que le slug de l'attribut est correct
```

```
        // Ajouter un message de débogage
```

```
        error_log('Type de transaction: ' . $type_of_transaction);
```

```
        if ($type_of_transaction == 'Vente') {
```

```
            echo '<p>Paiement par carte bancaire.</p>';
```

```
        } elseif ($type_of_transaction == 'Location (mensuel)') {
```

```
            echo '<p>Paiement par prélèvement automatique. Une caution de 30% du prix de vente sera prélevée.</p>';
```

```
        }
```

```
    }
```

```
}
```

//Afficher les informations de paiement sur la page de paiement : Ajouter un message personnalisé sur la page de paiement

```
add_action('woocommerce_review_order_before_payment', 'display_payment_info_on_checkout_page');
```

```
/**
```

```
 * Fonction pour afficher les informations de paiement sur la page de paiement.
```

```
 */
```

```
function display_payment_info_on_checkout_page()
```

```
{
```

```
    foreach (WC()->cart->get_cart() as $cart_item_key => $cart_item) {
```

```
        $product = $cart_item['data'];
```



```
if ($product->is_type('variation')) {
    $type_of_transaction = $product->get_attribute('pa_type_de_transaction'); // Assurez-vous
    que le slug de l'attribut est correct

    // Ajouter un message de débogage
    error_log('Type de transaction: ' . $type_of_transaction);

    if ($type_of_transaction == 'Vente') {
        echo '<p>Paielement par carte bancaire.</p>';
    } elseif ($type_of_transaction == 'Location (mensuel)') {
        echo '<p>Paielement par prélèvement automatique. Une caution de 30% du prix de vente sera
    prélevée.</p>';
    }
}
}
```

//Afficher la caution dans le panier (récapitulatif) : Ajouter un message personnalisé dans le récapitulatif du panier

```
add_action('woocommerce_cart_collaterals', 'display_deposit_message_in_cart_totals');
/**
 * Fonction pour afficher un message de caution dans le récapitulatif du panier.
 */
function display_deposit_message_in_cart_totals()
{
    foreach (WC()->cart->get_cart() as $cart_item_key => $cart_item) {
        $product = $cart_item['data'];
        if ($product->is_type('variation')) {
            $type_of_transaction = $product->get_attribute('pa_type_de_transaction'); // Assurez-vous
            que le slug de l'attribut est correct

            // Ajouter un message de débogage
            error_log('Type de transaction: ' . $type_of_transaction);

            if ($type_of_transaction == 'Location (mensuel)') {
                $original_price = $product->get_price() / 1.30; // Calculer le prix original
                $deposit = $original_price * 0.30; // 30% du prix de vente
                echo '<p>Une caution de ' . wc_price($deposit) . ' (30% du prix de vente) sera prélevée pour
            cette location.</p>';
            }
        }
    }
}
```



header.php :

```
<!DOCTYPE html>
<html <?php language_attributes(); ?>>

<head>
    <meta charset="<?php bloginfo('charset'); ?>">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!-- <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.15.4/js/all.js"
integrity="sha384-rOA1PnstxnOBLzCLMcre8ybwbtMemjzdNlLg8O7z1lUkLXozs4DHonlDtnE7fpc"
crossorigin="anonymous"> -->
    <!-- <script src="https://code.jquery.com/jquery-3.7.1.min.js" integrity="sha256-
/JqT3SQfawRcv/BIHPThkBs00EvtFFmqPF/ly/Cxo=" crossorigin="anonymous"></script> -->
    <!-- <script defer src="js/scripts.js"></script> -->
    <?php wp_head(); ?>
</head>

<body <?php body_class(); ?> <?php wp_body_open(); ?> <div id="wrapper">
    <header class="site-header">
        <a href="<?php echo esc_url(home_url()); ?>" title="<?php bloginfo('name'); ?>">
            <div id="logo">
                <?php $image = get_header_image(); ?>
                <?php if (!empty($image)) : ?>
                    height; ?>" width="<?php echo get_custom_header()->width; ?>" alt="logo" />
                <?php else : ?>
                    
                <?php endif; ?>
            </div>
            <?php if (display_header_text()) : ?>
                <div id="titre">
                    <h1 class="site-title"><?php bloginfo('name'); ?></h1>
                    <h2 class="site-description"><?php bloginfo('description'); ?></h2>
                </div>
            <?php endif; ?>
        </a>

        <!-- Icône du menu mobile -->
        <div id="mobile-menu-icon">
            <a class="toggle-nav" href="#"><i class="fas fa-bars"></i></a>
```



</div>

</header>

<nav id="menuPrincipal">

<!-- Menu Principal -->

<?php

wp_nav_menu(array(

'sort_column' => 'menu-order',

'theme_location' => 'principal',

'container' => false,

'menu_id' => 'main-menu',

));

?>

</nav>

<!--Fonction de recherche -->

<div id="recherche">

<i class="fas fa-search search-icon"></i>

<form role="search" method="get" id="searchform" action="<?php echo esc_url(home_url('/'));

?>">

<input type="text" value="<?php echo get_search_query(); ?>" name="s" id="s"

placeholder="Rechercher..." />

<input type="submit" id="searchsubmit" value="Rechercher" />

</form>

</div>

<?php do_action('after_main_menu'); ?>

<!-- Nous générons un id unique pour chaque contenu et nous ajoutons de la même manière que pour le body

des classes spécifiques générées automatiquement à chaque contenu. -->

<main id="post-<?php the_ID(); ?>" <?php post_class(); ?>>



Footer.php :

```
<footer>
<nav id="menuFooter">
    <?php wp_nav_menu(array(
        'sort_column' => 'menu_order',
        'theme_location' => 'footer',
        'fallback_cb' => false
    )); ?>
</nav>

<div id="footer_info">
    <div id="footer_left">

        <?php
        $tel = get_theme_mod('mzlleFlo_tel');
        if ($tel != '') {
            echo '<div id="tel"><i class="fas fa-phone-square-alt"></i> ' . $tel . '</div>';
        }
        $adress = get_theme_mod('mzlleFlo_adresse');
        if ($adress != '') {
            echo '<div id="adress"><i class="fas fa-map-marker-alt"></i> ' . $adress . '</div>';
        } ?>
    </div>
    <div id="footer_right">
        <?php
        $facebook = get_theme_mod('mzlleFlo_facebook');
        if ($facebook != '') {
            echo '<div id="facebook"><a href="' . $facebook . '"><i class="fab fa-
facebook"></i><p>Facebook</p></a></div>';
        }
        $tumblr = get_theme_mod('mzlleFlo_tumblr');
        if ($tumblr != '') {
            echo '<div id="tumblr"><a href="' . $tumblr . '"><i class="fab fa-tumblr-
square"></i><p>Tumblr</p></a></div>';
        }
        ?>

    </div>
</div>
<?php
```



```
$copyright = get_theme_mod('mzlleFlo_credits');
if ($copyright == 'oui') {
    echo '<div id="copyright"><i>&copy; M\' Zlle Flo - ' . date('Y') . ' / Tous droits réservés. Thème
par M\' Zlle Flo & Michel Grillon</i></div>';
} ?>
<a href="https://jigsaw.w3.org/css-validator/check/referer">
    
</a>
</footer>
</div>
<!--FERMETURE WRAPPER-->
<?php wp_footer(); ?>

<div class="fleche-hdpage" id="retour-haut">
    <i class="fas fa-arrow-up"></i>
</div>
</body>
</html>
```



index.php :

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

<?php
if (is_active_sidebar('sidebar_droite')) : ?>
    <section id="articles" class="haut avecAside">
        <?php else : ?>
            <section id="articles" class="haut">
                <?php endif;
                if (have_posts()) : while (have_posts()) : the_post(); ?>
                    <div class="unArticle">
                        <h2><a href="<?php the_permalink(); ?>"><?php the_title(); ?></a>
                        </h2>
                        <?php the_excerpt(); ?>
                        <p class="infos">
                            <?php comments_number('Aucun commentaire', 'un seul commentaire', '%
commentaires'); ?>
                        </p>
                    </div>
                <?php endwhile;

                if (empty(get_theme_mod('my_slider_first')) :
                    the_posts_pagination();
                endif;
            endif; ?>
        </section>

        <?php get_sidebar(); ?>

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




page-full.php :

```
<?php
/*
Template Name: Page sans sidebar
*/
get_header();
if (have_posts()) : while (have_posts()) : the_post();
?>
    <section id="pages" class="haut">
        <h2><?php the_title(); ?></h2>
        <?php the_content(); ?>
    </section>

<?php endwhile;
endif; ?>
</main>
<?php if (is_active_sidebar('sidebar-bas')) : ?>
    <aside id="bas">
        <?php dynamic_sidebar('sidebar-bas'); ?>
    </aside>
<?php endif;
get_footer(); ?>
```



plugin-art-gallery.php :

```
<?php
/*
Plugin Name: plugin-art-gallery
Plugin URI:
Description: Un plugin pour gérer et afficher une galerie d'art avec filtrage et défilement infini.
Author: Michel Grillon
Version: 1.0
Licence: GPL2
*/

// Enqueue scripts and styles for the frontend
add_action('wp_enqueue_scripts', 'enqueue_art_gallery_scripts');
function enqueue_art_gallery_scripts()
{
    // Vérifier si les styles Lightbox sont déjà enqueueés
    if (!wp_style_is('lightbox-css', 'enqueued')) {
        wp_enqueue_style('lightbox-css', plugin_dir_url(__FILE__) . 'css/lightbox.css');
    }

    // Vérifier si les scripts Lightbox sont déjà enqueueés
    if (!wp_script_is('lightbox-js', 'enqueued')) {
        wp_enqueue_script('lightbox-js', plugin_dir_url(__FILE__) . 'js/lightbox.js', array('jquery'), null,
true);
    }

    // Enqueue script de la galerie d'art
    wp_enqueue_script('art-gallery', plugin_dir_url(__FILE__) . 'js/art-gallery.js', array('jquery'), null,
true);

    // Localiser le script avec l'URL AJAX
    wp_localize_script('art-gallery', 'artGallery', array(
        'ajax_url' => admin_url('admin-ajax.php'),
        'nonce' => wp_create_nonce('load_more_artworks_nonce')
    ));
}

// Créer un type de publication personnalisé pour les œuvres d'art
add_action('init', 'create_art_post_type');
function create_art_post_type()
{

```



```
// Enregistrer le type de publication personnalisé 'artwork'
register_post_type('artwork', array(
    'labels' => array(
        'name' => __('Œuvres d\'art'), // Nom au pluriel
        'singular_name' => __('Œuvre d\'art') // Nom au singulier
    ),
    'public' => true, // Accessible publiquement
    'has_archive' => true, // A une archive
    'supports' => array('title', 'editor', 'thumbnail'), // Supporte le titre, l'éditeur et l'image à la une
    'show_in_rest' => true, // Disponible dans l'API REST
));
}

// Ajouter une meta box pour lier les œuvres d'art aux produits WooCommerce
add_action('add_meta_boxes', 'add_artwork_product_meta_box');
function add_artwork_product_meta_box()
{
    add_meta_box(
        'artwork_product_meta_box', // ID de la meta box
        __('Produit WooCommerce associé', 'textdomain'), // Titre de la meta box
        'display_artwork_product_meta_box', // Fonction de rappel pour afficher la meta box
        'artwork', // Type de publication auquel la meta box est ajoutée
        'side', // Position de la meta box (côté)
        'default' // Priorité de la meta box
    );
}

function display_artwork_product_meta_box($post)
{
    // Récupérer l'ID du produit WooCommerce associé à l'œuvre d'art
    $product_id = get_post_meta($post->ID, '_artwork_product_id', true);

    // Afficher le champ de sélection pour lier l'œuvre d'art à un produit WooCommerce
    echo '<label for="artwork_product_id">' . __('Sélectionner le produit WooCommerce', 'textdomain') .
    '</label>';
    echo '<select name="artwork_product_id" id="artwork_product_id">';
    echo '<option value="">' . __('Aucun', 'textdomain') . '</option>';

    // Récupérer tous les produits WooCommerce
    $products = wc_get_products(array('limit' => -1));
    foreach ($products as $product) {
        // Afficher chaque produit dans le champ de sélection
        echo '<option value="' . $product->get_id() . '">' . selected($product_id, $product->get_id(), false) .
    }
}
```



```
'> . $product->get_name() . '</option>';
}
echo '</select>';
}

// Sauvegarder la meta box lors de la sauvegarde de la publication
add_action('save_post', 'save_artwork_product_meta_box');
function save_artwork_product_meta_box($post_id)
{
    // Vérifier si l'ID du produit WooCommerce est défini
    if (isset($_POST['artwork_product_id'])) {
        // Mettre à jour la meta donnée avec l'ID du produit WooCommerce
        update_post_meta($post_id, '_artwork_product_id',
        sanitize_text_field($_POST['artwork_product_id']));
    }
}

// Créer un produit WooCommerce lorsque l'œuvre d'art est publiée
add_action('publish_artwork', 'create_woocommerce_product_from_artwork');
function create_woocommerce_product_from_artwork($post_id)
{
    // Vérifier si le type de publication est 'artwork'
    if (get_post_type($post_id) != 'artwork') {
        return;
    }

    // Récupérer l'ID du produit WooCommerce associé à l'œuvre d'art
    $product_id = get_post_meta($post_id, '_artwork_product_id', true);

    // Si aucun produit WooCommerce n'est associé, créer un nouveau produit
    if (empty($product_id)) {
        $post = get_post($post_id);
        $title = $post->post_title;
        $content = $post->post_content;
        $image_id = get_post_thumbnail_id($post_id);

        // Créer un nouveau produit WooCommerce
        $product = new WC_Product_Simple();
        $product->set_name($title);
        $product->set_description($content);
        $product->set_status('publish');
        $product->set_catalog_visibility('visible');
```



```
// Si une image à la une est définie, l'associer au produit
if ($image_id) {
    $product->set_image_id($image_id);
}

// Sauvegarder le produit et mettre à jour la meta donnée avec l'ID du nouveau produit
$new_product_id = $product->save();
update_post_meta($post_id, '_artwork_product_id', $new_product_id);
}
}

// Mettre à jour le produit WooCommerce lorsque l'œuvre d'art est mise à jour
add_action('save_post', 'update_woocommerce_product_from_artwork', 20, 1);
function update_woocommerce_product_from_artwork($post_id)
{
    // Vérifier si le type de publication est 'artwork'
    if (get_post_type($post_id) != 'artwork') {
        return;
    }

    // Récupérer l'ID du produit WooCommerce associé à l'œuvre d'art
    $product_id = get_post_meta($post_id, '_artwork_product_id', true);

    // Si un produit WooCommerce est associé, mettre à jour ses informations
    if (!empty($product_id)) {
        $post = get_post($post_id);
        $title = $post->post_title;
        $content = $post->post_content;
        $image_id = get_post_thumbnail_id($post_id);

        // Récupérer le produit WooCommerce
        $product = wc_get_product($product_id);
        if ($product) {
            // Mettre à jour les informations du produit
            $product->set_name($title);
            $product->set_description($content);

            // Si une image à la une est définie, l'associer au produit
            if ($image_id) {
                $product->set_image_id($image_id);
            }

            // Sauvegarder le produit
        }
    }
}
```



```
$product->save();
    }
}
}

// Charger les œuvres d'art supplémentaires pour le chargement infini via Ajax
add_action('wp_ajax_load_more_artworks', 'load_more_artworks');
add_action('wp_ajax_nopriv_load_more_artworks', 'load_more_artworks');

function load_more_artworks()
{
    check_ajax_referer('load_more_artworks_nonce', 'nonce');

    $page = isset($_POST['page']) ? intval($_POST['page']) : 1;
    $category = isset($_POST['category']) ? sanitize_text_field($_POST['category']) : '';

    $args = array(
        'post_type' => 'artwork',
        'posts_per_page' => get_option('posts_per_page'),
        'paged' => $page,
    );

    if ($category) {
        $args['tax_query'] = array(
            array(
                'taxonomy' => 'category',
                'field' => 'slug',
                'terms' => $category,
            ),
        );
    }

    $query = new WP_Query($args);

    if ($query->have_posts()) {
        while ($query->have_posts()) {
            $query->the_post();
            $product_id = get_post_meta(get_the_ID(), '_artwork_product_id', true);
            $artwork_id = get_post_meta($product_id, '_artwork_id', true);
            $artwork = get_post($artwork_id);

            if ($artwork) {
?>
```



```
<div class="art-gallery-item <?php echo strtolower(get_the_terms($artwork_id,
'category')[0]->name); ?>">
    <a href="<?php echo get_the_post_thumbnail_url($artwork_id, 'large'); ?>"
class="lightbox">
        <?php echo get_the_post_thumbnail($artwork_id, 'large'); ?>
        <h2><?php echo get_the_title($artwork_id); ?></h2>
        <p><?php echo get_post_meta($artwork_id, 'price', true); ?></p>
    </a>
</div>

<?php
    } else {
        wc_get_template_part('content', 'product');
    }
}
wp_reset_postdata();
}
wp_die();
}
```

```
// Shortcode pour afficher la galerie d'art
add_shortcode('art_gallery', 'display_art_gallery');
function display_art_gallery()
{
    ob_start();
    ?>
    <div class="art-gallery">
        <?php
            $args = array(
                'post_type' => 'artwork',
                'posts_per_page' => get_option('posts_per_page'),
                'paged' => 1,
            );
            $query = new WP_Query($args);

            if ($query->have_posts()) {
                while ($query->have_posts()) {
                    $query->the_post();
                    $product_id = get_post_meta(get_the_ID(), '_artwork_product_id', true);
                    ?>
                    <div class="art-gallery-item">
                        <a href="<?php echo get_the_post_thumbnail_url(get_the_ID(), 'large'); ?>"
class="lightbox">
                            <?php the_post_thumbnail('large'); ?>
```



```
<h2><?php the_title(); ?></h2>
<p><?php echo get_post_meta($product_id, '_price', true); ?></p>
</a>
</div>
<?php
}
wp_reset_postdata();
}
?>
</div>
<button id="load-more" data-next-page="2"><?php _e('Charger plus', 'textdomain'); ?></button>
<?php
return ob_get_clean();
}
?>
```




plugin-advanced-contact.php :

```
<?php
/*
Plugin Name: plugin-advanced-contact
Plugin URI:
Description:
Author: Michel Grillon
Version: 1.0
Licence: GPL2
*/

// Créer un type de publication personnalisé pour les soumissions de contact
add_action('init', 'create_contact_submissioncpt');
function create_contact_submissioncpt()
{
    // Enregistrer le type de publication personnalisé 'contact_submission'
    register_post_type('contact_submission', array(
        'labels' => array(
            'name' => __('Contact Submissions'), // Nom au pluriel
            'singular_name' => __('Contact Submission'), // Nom au singulier
        ),
        'public' => false, // Non accessible publiquement
        'show_ui' => true, // Afficher l'interface utilisateur
        'capability_type' => 'post', // Utiliser les capacités de type 'post'
        'supports' => array('title', 'editor', 'custom-fields'), // Supporte le titre, l'éditeur et les champs personnalisés
    ));
}

// Créer un shortcode pour le formulaire de contact
add_shortcode('advanced_contact_form', 'advanced_contact_form');
function advanced_contact_form()
{
    // Démarrer la mise en tampon de sortie
    ob_start(); ?>
    <form action="<?php echo esc_url(admin_url('admin-post.php')); ?>" method="post">
        <p>
            <label for="acf_name"><?php _e('Nom'); ?></label>
            <input type="text" id="acf_name" name="acf_name" required>
        </p>
        <p>
```



```
<label for="acf_email"><?php _e('Email'); ?></label>
<input type="email" id="acf_email" name="acf_email" required>
</p>
<p>
<label for="acf_message"><?php _e('Message'); ?></label>
<textarea id="acf_message" name="acf_message" required></textarea>
</p>
<input type="hidden" name="action" value="acf_contact_form">
<p>
<button type="submit"><?php _e('Envoyer'); ?></button>
</p>
</form>
<?php return ob_get_clean(); // Retourner le contenu mis en tampon
}

// Gérer les soumissions du formulaire de contact
add_action('admin_post_nopriv_acf_contact_form', 'acf_contact_form_handler');
add_action('admin_post_acf_contact_form', 'acf_contact_form_handler');
function acf_contact_form_handler()
{
    // Vérifier si les champs du formulaire sont définis
    if (isset($_POST['acf_name']) && isset($_POST['acf_email']) && isset($_POST['acf_message'])) {
        // Nettoyer et valider les données du formulaire
        $name = sanitize_text_field($_POST['acf_name']);
        $email = sanitize_email($_POST['acf_email']);
        $message = sanitize_textarea_field($_POST['acf_message']);

        // Créer une nouvelle publication de type 'contact_submission'
        $post_id = wp_insert_post(array(
            'post_title' => $name, // Titre de la publication
            'post_content' => $message, // Contenu de la publication
            'post_type' => 'contact_submission', // Type de publication
            'post_status' => 'publish', // Statut de la publication
            'meta_input' => array(
                'email' => $email // Meta donnée pour l'email
            )
        ));
        // Rediriger après la soumission
        wp_redirect(home_url('/thank-you'));
        exit;
    }
}
?>
```