

WordPress : Méta données

Pour manipuler les **méta données (post metas)** WordPress issues des **métabox**, vous allez créer une metabox personnalisée pour ajouter des champs supplémentaires à vos articles, pages ou types de publications personnalisés.

Une fois que les utilisateurs auront rempli ces champs dans l'éditeur, vous pourrez enregistrer ces données personnalisées en utilisant **update_post_meta** lorsque le contenu est sauvegardé.

Créer la metabox personnalisée avec add_meta_box :

Utilisez la fonction **add_meta_box** pour créer votre méta box personnalisée. Cette fonction vous permettra d'ajouter des champs supplémentaires dans l'éditeur de contenu où les utilisateurs pourront saisir des données spécifiques.

```
function mon_meta_box_callback( $post ) {
    // code pour afficher les champs de la méta box et récupérer les valeurs
    existantes si nécessaire
    $valeur_champ = get_post_meta( $post->ID, 'mon_champ', true );

    // Afficher le champ et utiliser la valeur existante si disponible
    echo '<input type="text" name="mon_champ" value="' . esc_attr(
    $valeur_champ ) . '" />';
}

function mon_meta_box() {
    add_meta_box( 'mon_meta_box_id', 'Ma Méta Box', 'mon_meta_box_callback',
    'post', 'normal', 'default' );
}

add_action( 'add_meta_boxes', 'mon_meta_box' );
```

Dans la fonction **mon_meta_box_callback**, vous devez afficher les champs HTML que vous souhaitez ajouter à la metabox.

Vous pouvez également récupérer les valeurs existantes des champs en utilisant **get_post_meta**.

Remarque : si le dernier paramètre est défini sur **true**, la fonction **get_post_meta** renverra uniquement la première valeur trouvée, même si plusieurs valeurs sont associées à la même clé.

Sauvegarder les données de la metabox avec save_post et update_post_meta :

Utilisez le hook **save_post** pour sauvegarder les données de la metabox lorsque le contenu est enregistré.

```
function mon_save_meta_box( $post_id ) {
    if ( defined( 'DOING_AUTOSAVE' ) && DOING_AUTOSAVE )
        return;

    if ( isset( $_POST['mon_champ'] ) ) {
        $valeur_champ = sanitize_text_field( $_POST['mon_champ'] );
```

```
        update_post_meta( $post_id, 'mon_champ', $valeur_champ );  
    }  
}
```

```
add_action( 'save_post', 'mon_save_meta_box' );
```

Dans cette fonction, nous utilisons **update_post_meta** pour sauvegarder la valeur du champ personnalisé (mon_champ) dans la base de données.

Remarque : c'est sur ce principe que sont enregistrées les valeurs des **champs ACF (Advanced Custom Fields)**.

Nous utilisons **sanitize_text_field** pour nettoyer la valeur avant de la sauvegarder pour des raisons de sécurité.

Remarque : en base de données, les méta données sont enregistrées dans la table **wp_postmeta**.