

Cette page explique comment incorporer des formulaires dans votre wiki. Les formulaires ne traitent pas les données entrées par les visiteurs, mais des modules externes peuvent le faire.

Balises

Deux directives sont utilisées pour commencer et terminer un formulaire :

```
(:input form "adresse" méthode:) ... (:input end:)
```

La directive `(:input form:)` commence un formulaire qui sera envoyé à *adresse* par la méthode *méthode*.

- L'*adresse* doit être entre guillemets.
- La *méthode* peut être "POST" (par défaut) ou "GET".

La directive `(:input end:)` ferme le formulaire.

Cette fonctionnalité avancée de [Pm Wiki](#) n'assure pas que le code HTML sera correct, elle assume que l'auteur sait ce qu'il fait. Notamment, `(:input form:)` et `(:input end:)` ne devraient pas être dans des tableaux, et tous les champs de formulaire devraient se trouver à l'intérieur d'un bloc `(:input form:)...(:input end:)`.

Champs du formulaire

Les champs d'un formulaire sont :

```
(:input text nom valeur size=n:) (:input hidden nom valeur:) (:input password  
nom valeur:) (:input radio nom valeur:) (:input checkbox nom valeur:) (:input  
select nom valeur étiquette:) (:input default nom valeur-par-défaut:) (:input  
submit nom valeur:) (:input textarea nom [=valeur=] rows=n cols=n:) (:input reset nom  
valeur:) (:input file nom étiquette:) (:input image nom "adresse" texte-alt.)
```

Les paramètres *nom* et *valeur* seront les attributs du code HTML : `name="nom"` et `value="valeur"`.

Pour la plupart des champs, la balise est du type :

```
(:input type nom valeur paramètre=valeur:)
```

où *type* est le type de l'élément, *nom* est le nom du champ, *valeur* est la valeur initiale et les paramètres spécifient des attributs additionnels au champ. Si la *valeur* contient des espaces, elle doit être entourée de guillemets; si elle contient des sauts de ligne (pour les champs "hidden" et "textarea"), elle doit être entourée de `[= . . . =]`.

Par exemple, voici comment créer un champ de texte avec une largeur de 30 caractères :

```
(:input text auteur "Patrick Michaud" size=30:)
```

Un auteur peut aussi spécifier le nom et la valeur d'un formulaire directement en utilisant les attributs `name=` et `value=` (comme dans le code HTML).

```
(:input text name=auteur value="Patrick Michaud" size=30:)
```

Le champ `textarea` (large aire de texte) peut avoir une valeur prédéfinie -- entourez la valeur de [= . . . =] si elle contient des espaces ou des sauts de ligne.

Le bouton `submit` est souvent écrit comme :

```
(:input submit value=étiquette:)
```

Voici un exemple plus avancé, formulaire de connexion :

```
(:input form "http://www.exemple.com":) (:input hidden action login:) || Utilisateur: || (:input text username:) || || Mot de passe: || (:input password password:) || || || (:input submit value="Connexion":) || (:input end:)
```

Utilisateur:

Mot de passe:

(:input select ... :)

Un champ de type "select" affiche une liste déroulante avec plusieurs options :

```
(:input form:) (:input select liste 1 "Un" :)
```



```
(:input select liste 2 "Deux" :)
```

```
(:input select liste 3 "Trois" :)
```

```
(:input submit value="OK":) (:input end:)
```

Les paramètres peuvent être spécifiés aussi comme :

```
(:input select name=liste value=1 label="Un" :)
```

Dans le premier élément, nous pouvons spécifier une hauteur de la boîte de sélection :

```
(:input select liste 1 "Un" size=3:)
```

Pour permettre la sélection de plusieurs options, le paramètre "multiple":

```
(:input select liste 1 "Un" size=3 multiple:)
```

Pour spécifier un élément qui sera sélectionné initialement :

```
(:input select liste 2 "Deux" selected=selected:)
```

Pour avoir deux listes déroulantes l'une à côté de l'autre, il faut leur donner un *nom* différent, mais aussi placer un séparateur entre elles -- un caractère, l'espace insécable ` ` ou encore l'espace `[=]` vide.

```
(:input form:) (:input select Liste1 1:)(:input select Liste1 2:)[ = ] (:input  
select Liste2 3:)(:input select Liste2 4:) (:input end:)
```

Voir aussi

(Pages en anglais)

- [Cookbook:Input Default](#) -- comment utiliser le type spécial *default* pour prédéfinir des valeurs
- [Cookbook:Form Validation](#) -- validation de données soumises par formulaire
- [Cookbook:Form Extensions](#) -- comment ajouter les balises *fieldset*, *legend*, et *label*.
- [Cookbook:Input Forms and JavaScript](#) -- [Java Script?](#) et les formulaires wiki.

Modules externes :

- [Cookbook:PmForm](#)
- [Cookbook:Fox](#)
- [Cookbook:Wiki Forms](#)
- [Cookbook:ProcessForm](#)

<< [Expressions fonctionnelles](#) | [Index de la documentation](#) | [Éditions simultanées](#) >>