

From EULER Project

PmWiki: Forms

This page explains how you can embed input forms into wiki pages.

Input forms don't actually handle processing of the form data -- the feature simply allows creation of forms inside wiki pages. Forms processing can be found in the Cookbook (see below).

Markup

Two directives are used to begin and end forms:

```
(:input form "url" method:)
...
(:input end:)
```

The `(:input form:)` directive starts a form that will post to *url* using the supplied *method*. The *url* must be in quotes. If the url is omitted, then the current page is assumed. If *method* is omitted then "POST" is assumed. The `(:input end:)` directive ends the current form.

Note that this feature doesn't ensure that the form output is correct HTML -- it assumes the author knows a little bit of what he or she is doing. Notably, `(:input form:)` and `(:input end:)` shouldn't appear inside tables, and all form fields and controls should be inside an `(:input form:)...(:input end:)` block.

Standard input controls

The standard input controls are:

```
(:input text name value size=n:)
(:input hidden name value:)
(:input password name value:)
(:input radio name value:)
(:input checkbox name value:)
(:input select name value label:)
(:input default default-name default-value:)
(:input submit name value:)
(:input textarea name value rows=n cols=n:)
(:input reset name label:)
(:input file name label:)
(:input image name "src" alt:)
```

Where *name* and *value* are in the HTML syntax: `name="addr" value="808 W Franklin"`.

For most controls the markup has the form:

```
(:input type name value parameter=value:)
```

where *type* is the type of input element (described below), *name* is the name of the control, *value* is its initial value, and parameters are used to specify additional attributes to the control. For example, the following creates a text input control with a size of 30 characters:

EULER Project | PmWiki / Forms

```
(:input text authorid "Jane Doe" size=30:)
```

For convenience, an author can also specify name and value arguments directly using name= and value= attributes (same as HTML):

```
(:input text name=authorid value="Jane Doe" size=30:)
```

For the textarea control a value can be set from PmWiki 2.2.0beta45 onwards.

The submit control will more often be written as:

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

Here's a more complete example, e.g., for a login prompt:

```
(:input form "http://www.example.com":)
(:input hidden action login:)
||      Name:||(:input text username:)      ||
|| Password:||(:input password password:)||
|| ||(:input submit value="Log In":) ||
(:input end:)
```

Name:

Password:

(:input select ... :)

The basic form of a select box is a sequence of options:

```
(:input form:)
(:input select name=abc value=1 label=alpha : )
(:input select name=abc value=2 label=beta : )
(:input select name=abc value=3 label=gamma : )
(:input submit:)
(:input end:)
```

The values can be specified positionally:

```
(:input select abc 1 alpha :)
```

We can specify the size of the selection box:

```
(:input select abc 1 alpha size=3 :)
```

You can specify a multiple select box:

```
(:input select abc 1 alpha size=3 multiple:)
```

To have an element selected, use selected=selected:

```
(:input select abc 2 beta selected=selected:)
```

Note that to have two select boxes inline, not only should you give them different `name=` parameters, but also place a separator, like a character, ` `; or even the null sequence `[==]` between them:

```
(:input form:)
(:input select name=FIRST value=1:)(:input select name=FIRST value=2:)[==]
(:input select name=SECOND value=3:)(:input select name=SECOND value=4:)
(:input end:)
```

See Also

- [Cookbook:Input Default](#)
- [Cookbook:Form Validation](#)
- [Cookbook:Form Extensions](#)
- [Cookbook:Input Forms and JavaScript](#)

Compatible recipes:

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

Retrieved from <https://www-sop.inria.fr/mascotte/EULER/wiki/pmwiki.php/PmWiki/Forms>
Page last modified on March 28, 2009, at 08:28 PM