From EULER Project

PmWiki: Other Variables

`$FmtV`

This variable is an array that is used for string substitutions at the end of a call to `FmtPageName()`. For each element in the array, the "key" (interpreted as a string) will be replaced by the corresponding "value". The variable is intended to be a place to store substitution variables that have frequently changing values (thus avoiding a rebuild of the variable cache making `FmtPageName()` faster). Also see `$FmtP`. *Values of `$FmtV` are set by the internal functions FormatTableRow, LinkIMap, HandleBrowse, PreviewPage, HandleEdit, PmWikiAuth, and PasswdVar, apparently to set values for system generated string substitutions like PageText.*

`$FmtP`

This variable is an array that is used for pattern substitutions near the beginning of a call to `FmtPageName`. For each element in the array, the "key" (interpreted as a pattern) will be replaced by the corresponding value evaluated for the name of the current page. This is for instance used to handle $-substitutions that depend on the pagename passed to `FmtPageName()`. Also see `$FmtV`. From robots.php: *If $EnableRobotCloakActions is set, then a pattern is added to `$FmtP` to hide any "?action=" url parameters in page urls generated by PmWiki for actions that robots aren't allowed to access. This can greatly reduce the load on the server by not providing the robot with links to pages that it will be forbidden to index anyway.*

`$FmtPV`

This variable is an array that is used for defining Page Variables. New variables can be defined with `$FmtPV['$VarName'] = 'variable definition';` which can be used in markup with `{$VarName}`. Please note that the contents of `$FmtPV['$VarName']` are `eval()`ed to produce the final text for `$VarName`, so the contents must be a PHP expression which is valid at the time of substitution. In particular, this does not work:

```
#This doesn't work
$FmtPV['$MyText'] = "This is my text."; # WARNING: Doesn't work!
```

The problem is that the text `This is my text.` is not a valid PHP expression. To work it would need to be placed in quotes, so that what actually gets stored in `$FmtPV['$MyText']` is `"This is my text."` which **is** a valid PHP expression for a text string. Thus the correct way to do this would be with an extra set of quotes:

```
#This will work
$FmtPV['$MyText'] = '"This is my text."';
```

This also has implications for how internal PHP or PmWiki variables are accessed. To have the page variable `$MyVar` produce the contents of the internal variable `$myvar`, many folks try the following which does not work:

```
#This doesn't work either!
$myvar = SomeComplexFunction();
$FmtPV['$MyVar'] = $myvar; # WARNING: Doesn't work!
```

There are several correct ways to do this, depending on whether you need the value of the `$myvar` variable as it was at the time the `$FmtPV` entry was created, or at the time that a particular instance of `$MyVar` is being rendered on a page. For most simple page variables that don't change during the processing of a page its more efficient to set the value when the entry is created:

```
$myvar = SomeComplexFunction();
$FmtPV['$MyVar'] = "'" . $myvar . "'"; #capture contents of
$myvar
```

NOTE: If `$myvar` should contain single quotes, the above won't work as is, and you'll need to process the variable to escape any internal quotes.

For more complex cases where an internal variable may have different values at different places in the page (possibly due to the effects of other markup), then you need to make the `$FmtPV` entry make an explicit reference to the global value of the variable (and the variable had better be global) like this:

```
global $myvar;
$FmtPV['$MyVar'] = '$GLOBALS["myvar"]';
```

Finally, there's nothing to stop you from simply having the evaluation of the `$FmtPV` entry execute a function to determine the replacement text:

```
# add page variable {$Today}, formats today's date as
yyyy-mm-dd
$FmtPV['$Today'] = 'strftime("%Y-%m-%d", time() )';
```

See Cookbook:MoreCustomPageVariables for more examples of how to use `$FmtPV`.

`$MaxPageTextVars`

This variable prevents endless loops in accidental recursive PageTextVariables which could lock down a server. Default is 500 which means that each PageTextVariable from one page can be displayed up to 500 times in one wiki page.

Retrieved from https://www-sop.inria.fr/mascotte/EULER/wiki/pmwiki.php/PmWiki/OtherVariables
Page last modified on July 05, 2009, at 03:26 PM