

From EULER Project

PmWiki: Markup Expressions

The `{ ( . . . ) }` "expression markup" allows for a variety of string and formatting operations to be performed from within markup. Operations defined by this recipe include *substr*, *ftime*, *strlen*, *rand*, *toupper* / *tolower*, *ucfirst*, *ucwords*, *pagename* and *asspaced*.

substr

The "substr" expression extracts portions of a string. The arguments are

1. the string to be processed. Always quote the string to be processed.
2. the initial position of the substring. Note that the initial position argument is zero-based (i.e., the first character is referenced via a "0").
3. the number of characters to extract

```
{ (substr "PmWiki" 2 3) } Wik
{ (substr "PmWiki" 2) } Wiki
{ (substr "PmWiki" 0 1) } P
{ (substr "PmWiki" 0 -3) } PmW
{ (substr "PmWiki" -3) } iki
```

To obtain the last *n* characters of a string use `{ (substr "string" -n) }`

To truncate the last *n* characters of a string use `(substr "string" 0 -n) }`

ftime

"Ftime" expressions are used for date and time formatting. The generic form is

```
{ (ftime "fmt" "when" ) }
{ (ftime fmt="fmt" when="when" ) }
```

where *fmt* is a formatting string and *when* is the time to be formatted. The arguments can be in either order and may use the optional "fmt=" and "when=" labels.

Examples:

```
{ (ftime) } April 18, 2024, at 05:13 AM
{ (ftime fmt="%F %H:%M") } 2024-04-18 05:13
{ (ftime %Y) } 2024
{ (ftime fmt=%T) } 05:13:26
{ (ftime when=tomorrow) } April 19, 2024, at 12:00 AM
{ (ftime fmt="%Y-%m-%d" yesterday) } 2024-04-17
{ (ftime week %F) } 1970-01-01
{ (ftime fmt=%D month) } 01/01/70
{ (ftime fmt="%a%e %b" when="next week" ) } Thu25 Apr
```

The *fmt* parameter is whatever is given by "fmt=", the first parameter containing a '%', or else the site's default. The formatting codes are described at <http://php.net/strftime>. In addition to those, 's' produces Unix timestamps. Some common formatting strings:

EULER Project | PmWiki / Markup Expressions

%F	# ISO-8601 dates	"2024-04-18"
%s	# Unix timestamp	"1713410006"
%H:%M:%S	# time as hh:mm:ss	"05:13:26"
%m/%d/%Y	# date as mm/dd/yyyy	"04/18/2024"
"%A, %B %d, %Y"	# in words	"Thursday, April 18, 2024"

The *when* parameter understands many different date formats. The *when* parameter is whatever is given by "when=", or whatever parameter remains after determining the format parameter. Some examples:

2007-04-11	# ISO-8601 dates
20070411	# dates without hyphens, slashes, or dots
2007-03	# months
@1176304315	# Unix timestamps (seconds since 1-Jan-1970 00:00 UTC)
now	# the current time
today	# today @ 00:00:00
yesterday	# yesterday @ 00:00:00
"next Monday"	# relative dates
"last Thursday"	# relative dates
"-3 days"	# three days ago
"+2 weeks"	# two weeks from now

Note: If you want to convert a Unix timestamp you **must** prefix with the @. Thus, "{ (ftime "%A, %B %d, %Y" @1231116927) }".

The *when* parameter uses PHP's `strtotime()` function to convert date strings according to the GNU date input formats; as of this writing it only understands English phrases in date specifications.

The variable `$FTimeFmt` can be used to override the default date format used by the "ftime" function. The default `$FTimeFmt` is `$TimeFmt`.

strlen

The "strlen" expression returns the length of a string. The first argument is the string to be measured.

```
{ (strlen "{$:Summary}") } 32
```

rand

The "rand" expression returns a random integer. The first argument is the minimum number to be returned and the second argument is the maximum number to be returned. If called without the optional min, max arguments `rand()` returns a pseudo-random integer between 0 and `RAND_MAX`. If you want a random number between 5 and 15 (inclusive), for example, use `rand (5, 15)`.

```
{ (rand) } 225081062
```

toupper / tolower

The "toupper" and "tolower" expressions convert a string into uppercase or lowercase. The first argument is the string to be processed.

```
{(toupper "{$:Summary}")}  STRING AND FORMATTING OPERATIONS  
{(tolower "{$:Summary"})}  string and formatting operations
```

ucfirst

The "ucfirst" expression converts the first character of a string to uppercase. The first argument is the string to be processed.

```
{(ucfirst "{$:Summary"})}  String and formatting operations
```

ucwords

The "ucwords" expression converts the first character of each word in a string to uppercase. The first argument is the string to be processed.

```
{(ucwords "{$:Summary"})}  String And Formatting Operations
```

pagename

The "pagename" expression builds a pagename from a string. The first argument is the string to be processed.

```
{(pagename "{$:Summary"})}  PmWiki.StringAndFormattingOperations
```

asspaced

The "asspaced" expression formats wikiwords. The first argument is the string to be processed.

```
{(asspaced "{$FullName"})}  Pm Wiki.Markup Expressions
```

Nesting expressions

Markup expressions can be nested:

```
{(tolower (substr "Hello World" 2))}  llo world
```

Notes

- Some of the string-processing markups may not work properly on UTF-8 characters or escaped sequences.
- The ftime markup does not work with some ISO 8601([approve sites](#)) dates (because a time of 24:00 is invalid([approve sites](#)))such as:

```
{(ftime fmt="%m/%d/%Y @ %H:%M:%S" 06/27/2007 @ 00:00:00  
when="20070626T2400")} 00:00:00  
{(ftime fmt="%H:%M:%S"  
when="20070626T2400")}
```

- is it possible to display the time in another time zone([approve sites](#)), eg

```
NowTime: { (ftime fmt="%F %H:%M")}
{ (ftime when='TZ=:Europe/London" ')}
{ (ftime when='TZ=Europe/London" 0
days')}
{ (ftime when="TZ='Pacific/Auckland'
2004-10-31 08:30")}
{ (ftime when='TZ="Pacific/Auckland"
{$:NowTime}')} }
```

NowTime: 2024-04-18 05:13

January 01, 1970, at 01:00 AM
January 01, 1970, at 01:00 AM
October 31, 2004, at 12:00 AM
(ftime when='TZ=Pacific/Auckland {2024-04-18 05:13}')

See also

- [Page variables](#), [Page text variables](#)
- [Conditional markup](#)
- [Cookbook:MarkupExprPlus](#)

Retrieved from <https://www-sop.inria.fr/mascotte/EULER/wiki/pmwiki.php/PmWiki/MarkupExpressions>
Page last modified on May 19, 2009, at 03:36 AM