

# Macros

*Special text strings expand on the fly to display user data or system info*

Macros are text strings in one of two forms:

```
%MACRONAME%  
%MACRONAME{ parameter="value" }%
```

These usually expand into content when a topic is rendered for viewing. There are two types of macros:

1. [Preference settings](#): May be defined and modified by the user
2. Registered macros: Defined by the system or by Plugins (for example, the [SpreadSheetPlugin](#) introduces a `%CALC{ }%` macro)

## On this page:

- ↓ [Using Macros](#)
  - ↓ [Order of expansion](#)
    - ↓ [Standard form](#)
      - ↓ [Animated Example](#)
    - ↓ [Delayed form](#)
      - ↓ [Animated Example](#)
- ↓ [Macro Names](#)
  - ↓ [Parameters](#)
    - ↓ [Parameter defaults](#)
  - ↓ [Access Control Settings](#)
  - ↓ [Local values for preferences](#)
- ↓ [Predefined Macros](#)
  - ↓ [ACTIVATEDPLUGINS](#) -- list of currently activated plugins
  - ↓ [ADDTOHEAD](#)
  - ↓ [ADDTOZONE](#)
    - ↓ [What is a "Zone"?](#)
    - ↓ [Adding content to a zone](#)
    - ↓ [Enforcing a linear order of content within a zone](#)
    - ↓ [Working with {MergeHeadAndScriptZones} disabled \(default\)](#)
    - ↓ [Working with {MergeHeadAndScriptZones} enabled](#)
    - ↓ [Example: Adding to a zone with missing dependencies](#)
    - ↓ [Example: Adding Javascript to a page](#)
    - ↓ [Example: Adding CSS to a page](#)
  - ↓ [ALLVARIABLES](#) -- list of currently defined macros
  - ↓ [AQUA](#) -- start aqua colored text
  - ↓ [ATTACHURL](#) -- full URL for attachments in the current topic
  - ↓ [ATTACHURLPATH](#) -- path of the attachment URL of the current topic
  - ↓ [AUTHREALM](#) -- authentication realm
  - ↓ [BASETOPIC](#) -- base topic where an INCLUDE started
  - ↓ [BASEWEB](#) -- base web where an INCLUDE started
  - ↓ [BB](#) -- bullet with line break
  - ↓ [BB2](#) -- level 2 bullet with line break
  - ↓ [BB3](#) -- level 3 bullet with line break
  - ↓ [BB4](#) -- level 4 bullet with line break
  - ↓ [BLACK](#) -- start black colored text
  - ↓ [BLUE](#) -- start blue colored text
  - ↓ [BR](#) -- line break
  - ↓ [BROWN](#) -- start brown colored text
  - ↓ [BULLET](#) -- bullet character
  - ↓ [BUTTON{"text" ...}](#) -- renders a nice button


- ↓ CALC{"formula"} -- add spreadsheet calculations to tables and outside tables
- ↓ CARET -- caret symbol
- ↓ COMMENT{ attributes } -- insert an edit box into the topic to easily add comments.
- ↓ COVER -- current skin cover
- ↓ DATE -- signature format date
- ↓ DISPLAYTIME{"format"} -- formatted display time
- ↓ EDITACTION -- Selects an edit template
- ↓ EDITTABLE{ attributes } -- edit tables using edit fields and other input fields
- ↓ ENCODE{"string"} -- encodes a string
- ↓ ENDCOLOR -- end colored text
- ↓ ENDSECTION{"name"} -- marks the end of a named section within a topic
- ↓ ENDTAB -- ending marker for a tab of a tabpane
- ↓ ENDTABPANE -- ending tag for tabpane widget
- ↓ ENDTWISTY
- ↓ ENDTWISTYTOGGLE
- ↓ ENV{"varname"} -- inspect the value of an environment variable
- ↓ EXAMPLETAG -- example variable
- ↓ EXPAND{"expression" scope="topicexpandin" ...}%
- ↓ FAILEDPLUGINS -- debugging for plugins that failed to load, and handler list
- ↓ FORMAT{"list" format="" header="" footer="" separator=""} -- format a list of objects
  - ↓ Supported formatting tokens
- ↓ FORMFIELD{"fieldname"} -- renders a field in the form attached to some topic
- ↓ GENPDF -- Insert a link that generated a PDF for the current or identified topic
- ↓ GMTIME{"format"} -- formatted GM time
- ↓ GRAY -- start gray colored text
- ↓ GREEN -- start green colored text
- ↓ GROUPINFO{"name"} -- retrieve details about a group
- ↓ GROUPS -- a formatted list of groups
- ↓ H -- help icon
- ↓ HISTORY{ attributes } -- control attributes of tables and sorting of table columns
- ↓ Additional macros
- ↓ HOMETOPIC -- home topic in each web
- ↓ HTTP -- get HTTP headers
- ↓ HTTP\_HOST -- environment variable
- ↓ HTTPS -- get HTTPS headers
- ↓ I -- idea icon
- ↓ ICON{"name" alt="" default="name"} -- small documentation graphic or icon of common attachment types
- ↓ ICONURL{"name" default="name"} -- URL of small documentation graphic or icon
- ↓ ICONURLPATH{"name" default="name"} -- URL path of small documentation graphic or icon
- ↓ IF{"condition" ...} -- simple conditionals
- ↓ INCLUDE{"topic"} -- include other topic.
- ↓ INCLUDE{"url"} -- include a web page
- ↓ INCLUDE{"doc:"} -- include Foswiki embedded module documentation
- ↓ INCLUDINGTOPIC -- name of topic that includes current topic
- ↓ INCLUDINGWEB -- web that includes current topic
- ↓ JQICON{"name" ...} -- render an image
- ↓ JQICONPATH{"name"} -- render the urlpath to an image
- ↓ JQPLUGINS{"plugins" ... } -- display a summary of available plugins
- ↓ JQREQUIRE{"plugin, plugin, ... "} -- enable a plugin on the current page
- ↓ JQTHEME{"name" ...} -- switch jQuery UI theme
- ↓ LANG -- the lang attribute of generated HTML pages
- ↓ LANGUAGE -- current user's language
- ↓ LANGUAGES -- list available languages
- ↓ LIME -- start lime colored text
- ↓ LOCALSITEPREFS -- web.topicname of site preferences topic
- ↓ LOGIN -- present a full login link
- ↓ LOGOUT -- present a full logout link

- ↓ M -- moved to... icon
- ↓ MAINWEB -- synonym for USERSWEB
- ↓ MAKETEXT -- creates text using Foswiki's I18N infrastructure
- ↓ MAROON -- start maroon colored text
- ↓ META -- displays meta-data
- ↓ METASEARCH -- special search of meta data
- ↓ N -- "new" icon
- ↓ NAVY -- start navy colored text
- ↓ NOP -- template text not to be expanded in instantiated topics
- ↓ NOTIFYTOPIC -- name of the notify topic
- ↓ OLIVE -- start olive colored text
- ↓ ORANGE -- start orange colored text
- ↓ P -- pencil icon
- ↓ PINK -- start pink colored text
- ↓ PLUGINDESCRIPTIONS -- list of plugin descriptions
- ↓ PLUGINVERSION -- the version of a Foswiki Plugin, or the Foswiki Plugins API
- ↓ POPUPWINDOW{"topic" ...} -- opens a topic or url in a new window
- ↓ PUBURL -- the base URL of attachments
- ↓ PUBURLPATH -- the base URL path of attachments
- ↓ PURPLE -- start purple colored text
- ↓ Q -- question icon
- ↓ QUERY -- get the value of meta-data
- ↓ QUERYPARAMS -- show paramaters to the query
- ↓ QUERYSTRING -- full, unprocessed string of parameters to this URL
- ↓ RED -- start red colored text
- ↓ REMOTE\_ADDR -- environment variable
- ↓ REMOTE\_PORT -- environment variable
- ↓ REMOTE\_USER -- environment variable
- ↓ RENDERLIST -- render bullet lists in a variety of formats
- ↓ RENDERZONE
- ↓ REVARG -- &rev=n URL revision parameter of current topic
- ↓ REVINFO -- revision information of current topic
- ↓ REVINFO{"format"} -- formatted revision information of topic
- ↓ REVTITLE -- (r1) The requested revision as displayed in topic breadcrumbs
- ↓ S -- red star icon
- ↓ SCRIPTNAME -- name of current script
- ↓ SCRIPTSUFFIX -- script suffix
- ↓ SCRIPTURL{"script"} -- URL of script
- ↓ SCRIPTURLPATH{"script"} -- URL path of script
- ↓ SEARCH{"text"} -- search content
- ↓ SERVETIME{"format"} -- formatted server time
- ↓ SESSIONID -- unique ID for this session
- ↓ SESSIONVAR -- name of CGI and session variable that stores the session ID
- ↓ SESSION\_VARIABLE -- get, set or clear a session variable
- ↓ SHOWPREFERENCE -- show where preferences are defined.
- ↓ SILVER -- start silver colored text
- ↓ SKIN -- current skin
- ↓ SLIDESHOWEND -- end slideshow
- ↓ SLIDESHOWSTART -- convert a topic with headings into a slideshow
- ↓ SPACEDTOPIC -- topic name, spaced and URL-encoded deprecated
- ↓ SPACEOUT{"string"} -- renders string with spaces inserted in sensible places
- ↓ STARTINCLUDE -- start position of topic text if included
- ↓ STARTSECTION -- marks the start of a section within a topic
- ↓ STATISTICSTOPIC -- name of statistics topic
- ↓ STOPINCLUDE -- end position of topic text if included
- ↓ SUBSCRIBE{ attributes } - subscribe the current user for notification of changes to the current topic
- ↓ SYSTEMWEB -- name of documentation web

- ↓ T -- tip icon
- ↓ TAB{"text" ...} -- tab inside a tabpane widget
- ↓ TABLE{ attributes } -- control attributes of tables and sorting of table columns
  - ↓ Attributes for tables
  - ↓ Attributes for table sorting
  - ↓ Attributes for table cells
  - ↓ Attributes for data cells
  - ↓ Attributes for headers
  - ↓ Other attributes
- ↓ TABPANE{...} -- tabpane widget
- ↓ TEAL -- start teal colored text
- ↓ TOC{"Topic"} -- table of contents
- ↓ TOPIC -- name of current topic
- ↓ TOPICLIST{"format"} -- topic index of a web
- ↓ TOPICURL -- shortcut to viewing the current topic
- ↓ TWIKIWEB -- synonym for SYSTEMWEB
- ↓ TWISTY
- ↓ TWISTYBUTTON
- ↓ TWISTYHIDE
- ↓ TWISTYSHOW
- ↓ TWISTYTOGGLE
- ↓ U -- "updated" icon
- ↓ URLPARAM{"name"} -- get value of a URL parameter
- ↓ USERINFO{"name" format=""} -- retrieve details about a user
- ↓ USERNAME -- your login username
- ↓ USERSWEB -- name of users web
- ↓ VAR{"NAME" web="Web"} -- get a preference value from another web
- ↓ VBAR -- vertical bar
- ↓ WEB -- name of current web
- ↓ WEBLIST{"format"} -- index of all webs
- ↓ WEBPREFSTOPIC -- name of web preferences topic
- ↓ WHITE -- start white colored text
- ↓ WIKIHOMEURL -- site home URL
- ↓ WIKINAME -- your Wiki username
- ↓ WIKIPREFSTOPIC -- name of site-wide preferences topic
- ↓ WIKITOOLNAME -- name of your site
- ↓ WIKIUSERNAME -- your Wiki username with web prefix
- ↓ WIKIUSERSTOPIC -- name of topic listing all registered users
- ↓ WIKIVERSION -- the version of the installed Foswiki engine
- ↓ WIKIWEBMASTER -- feedback email address for site
- ↓ WIKIWEBMASTERNAME -- Name of the administrator for the site
- ↓ X -- warning icon
- ↓ Y -- "yes" icon
- ↓ YELLOW -- start yellow colored text
- ↓ Shortcuts

## Using Macros

To use a macro type its name. For example,

- type %T% to get  (a [preference settings](#))
- type %TOPIC% to get `Macros` (a predefined [macro](#))
- type %CALC{ "\$UPPER(Text)" }% to get `TEXT` (`CALC` is a macro defined by [SpreadSheetPlugin](#))

### Note:


- To leave a macro unexpanded, precede it with an exclamation point, e.g. type `!%TOPIC%` to get `%TOPIC%`
  - Alternatively, insert a `<nop>` anywhere in the macro, Eg. `%<nop>TOPIC%`

- Macros are expanded relative to the topic they are *used* in, not the topic they are *defined* in
- Type `%ALLVARIABLES%` to get a full listing of all macros defined for a particular topic
- If a macro is not defined, then it will be left in the text *unless* it is called with a `default` parameter, in which case the value of the `default` parameter will replace the macro call in the output. For example, `%UNDEFINED{default="blank"}%` will expand to `blank`.

## Order of expansion

The following describes only these types of macros:

- [Preference settings](#)
- Most macros provided by plugins (those that use `registerTagHandler()`)

 **Not** those that use `commonTagsHandler()`

- Most built-in Foswiki macros

 Notable exceptions include: [CALC](#), [STARTSECTION/ENDSECTION](#), [STARTINCLUDE/STOPINCLUDE](#)

## Standard form

The key to understanding nested expressions in Foswiki is to understand that macros are expanded "inside-out, left-to-right". **Example:**

```
%MACRO1{
  something="%MACRO2{
    somethingelse="%MACRO3%, %MACRO4%"
  }%"
}%
```

The macros are expanded in this order: MACRO3, MACRO4, MACRO2, MACRO1.

## Animated Example

### Error: no such plugin CHILI

```
%INCLUDE{
  "%QUERY{
    "'%THETOPIC%'/%THEFIELD%"
  }%"
  section="Summary"
}%
* Set THETOPIC = %SYSTEMWEB%.FAQWhatIsWikiWiki
* Set THEFIELD = TopicClassification
```

```
%INCLUDE{
  "%QUERY{
    "'%SYSTEMWEB%.FAQWhatIsWikiWiki'/%THEFIELD%"
  }%"
  section="Summary"
}%
* Set THETOPIC = %SYSTEMWEB%.FAQWhatIsWikiWiki
* Set THEFIELD = TopicClassification
```

```
%INCLUDE{
  "%QUERY{
    "'%SYSTEMWEB%.FAQWhatIsWikiWiki'/TopicClassification"
  }%"
  section="Summary"
}%
* Set THETOPIC = %SYSTEMWEB%.FAQWhatIsWikiWiki
* Set THEFIELD = TopicClassification
```

```
%INCLUDE{
  "%QUERY{
    "'System.FAQWhatIsWikiWiki'/TopicClassification"
  }%"
  section="Summary"
}%
* Set THETOPIC = %SYSTEMWEB%.FAQWhatIsWikiWiki
* Set THEFIELD = TopicClassification
```

```
%INCLUDE{
  "FrequentlyAskedQuestion"
  section="Summary"
}%
* Set THETOPIC = %SYSTEMWEB%.FAQWhatIsWikiWiki
* Set THEFIELD = TopicClassification
```

These topics are for frequently asked questions including answers.


```
* Set THETOPIC = %SYSTEMWEB%.FAQWhatIsWikiWiki
* Set THEFIELD = TopicClassification
```

These topics are for frequently asked questions including answers.

```
* Set THETOPIC = System.FAQWhatIsWikiWiki
* Set THEFIELD = TopicClassification
```

## Delayed form

**Standard form** macros can nearly always be used to build the parameter string of another macro; however, sometimes it is desirable to bypass the **inside-out** expansion order and delay the inner macro until after the outer macro has finished expansion. This is accomplished by using the **\$percent** format token instead of **%**, and escaping any **"** character it uses (becomes **\"**)

 When working with a given **macro**, consult its documentation to determine which parameters support the **\$percent/\$percnt format tokens**. Generally only *output parameters* like **header**, **format** and **footer** support **format tokens**.

### Example:

```
%MACRO1{
  format="$percentMACRO2{
    format="\%MACRO3%, %MACRO4%"
  }$percent"
}%
```

The macros are expanded in this order: MACRO3, MACRO4, MACRO1, **MACRO2**.

## Animated Example

From the [conditional output example](#):

### Error: no such plugin CHILI

```
%SEARCH{
  "info.date >= d2n('2009-01-01') AND info.date <= d2n('2009-12-31')"
  type="query"
  limit="2"
  nonoise="on"
  format=" * $percentICON{
    \"$percentIF{
      \"'$topic'/parent.name='%PARENT%'\
      then=\"info\" else=\"gear\"
    }$percent\"
  }$percent [[ $topic ]]"
}%
----
* Set PARENT = UserDocumentationCategory
```

```
%SEARCH{
  "info.date >= d2n('2009-01-01') AND info.date <= d2n('2009-12-31')"
  type="query"
  limit="2"
  nonoise="on"
  format=" * $percentICON{
    \"$percentIF{
      \"'$topic'/parent.name='UserDocumentationCategory'\
      then=\"info\" else=\"gear\"
    }$percent\"
  }$percent [[ $topic ]]"
}%
----
* Set PARENT = UserDocumentationCategory
```

```
* %ICON{
  "%IF{
    "'AccessKeys'/parent.name='UserDocumentationCategory'"
    then="info" else="gear"
  }%"
}% [[AccessKeys]]
* %ICON{
  "%IF{
    "'AdminSkillsAssumptions'/parent.name='UserDocumentationCategory'"
    then="info" else="gear"
  }%"
}%
```

```
}% [[AdminSkillsAssumptions]]
----
* Set PARENT = UserDocumentationCategory
```

```
* %ICON{
  "info"
}% [[AccessKeys]]
* %ICON{
  "gear"
}% [[AdminSkillsAssumptions]]
----
* Set PARENT = UserDocumentationCategory
```

```
*  [[AccessKeys]]
*  [[AdminSkillsAssumptions]]
----
* Set PARENT = UserDocumentationCategory
```

See also: [FormattedSearch](#)

## Macro Names

Macro names must start with a letter. The following characters can be letters, numbers and the underscore '\_'. Letters may be upper or lower-case, E.g. %MYVAR%, %MyVar%, %My2ndVar%, and %My\_Var% are all separate, valid macro names (macros are case sensitive - %MyVAR% and %MYVAR% are not the same).

By convention all settings, predefined macros and macros registered by plugins are always UPPER-CASE.

## Preference Settings

A *preference setting* lets you define a simple [macro](#) that will be expanded in your output. A preference setting looks like this:

```
[multiple of 3 spaces] * [space] Set [space] MACRONAME [space] = [space] value
```

Example:

```
* Set WEBBGCOLOR = #FFFFC0
```

Macros defined using preference settings are expanded by enclosing their name in percent signs. So when you write %WEBBGCOLOR%, it gets expanded to #FFD8AA

A preference macro is always taken from the most current topic revision, even when accessing previous revisions of a topic.

Preferences can be defined in a number of places:

1. [DefaultPreferences](#) (Foswiki upgrades overwrite this topic)
2. [SitePreferences](#)
3. Sub-webs inherit the WebPreferences of their parent
4. WebPreferences
5. In user topics, if the user has one (yours is [Main.WikiGuest](#))
6. In (some) plugin documentation topics
7. In the topic being accessed

Set statements which occur at higher-numbered locations override macros of the same name defined at lower numbered levels, *unless the macro was listed in a FINALPREFERENCES setting (finalised) at a lower-*



*numbered level*. In this case, the macro is locked to the value at that level; `Set` statements at higher-numbered levels are ignored.

💡 Preference settings can easily be disabled with a `#` sign. Example:

```
* #Set DENYWEBCHANGE = %USERSWEB%.UnknownUser
```

💡 You can hide preference settings in the output by enclosing them in HTML comments; for example,

```
<!--  
  * Set HIDDEN = This will be invisible in the output  
-->
```

If you are setting a preference and using it in the same topic, note that Foswiki reads all the preference settings from the saved version of the topic before it displays anything. This means you can use a setting anywhere in the topic, even if you set it at the very end. **But beware:** it also means that if you change the setting of a macro you are using in the same topic, `Preview` will show the wrong thing, and you must `save` the topic to see it correctly.

Also note that Foswiki always reads the setting from the most current topic revision, so viewing older revisions of a topic can show unexpected results.

And especially important, **preference settings are never overridden or set in "%INCLUDE{" topics**. in the below example about weather conditions, note the difference in the `CONDITIONS` expansion

Spaces between the `=` sign and the value will be ignored. You can split a value over several lines by indenting following lines with spaces - as long as you don't try to use `*` as the first character on the following line.

**Example:**

```
* Set MACRONAME = value starts here  
  and continues here
```

Whatever you include in your macro will be expanded on display, exactly as if it had been entered directly (though see `Parameters`, below).

**Example: Create a custom logo macro**

- To place a logo anywhere in a web by typing `%MYLOGO%`, define the preference settings in the web's `WebPreferences` topic, and upload a logo file, ex: `mylogo.gif`. You can upload by [attaching the file to WebPreferences](#), or, to avoid clutter, to any other topic in the same web, e.g. `LogoTopic`. Sample preference setting in [WebPreferences](#):

```
* Set MYLOGO = %PUBURL%/ %WEB%/LogoTopic/mylogo.gif
```

You can also set preference settings in a topic by clicking the link `Edit topic preference settings` under `More topic actions`. Preferences set in this manner are known as 'meta' preferences and are not visible in the topic text, but take effect nevertheless.

## Parameters

Note that `%CONDITIONS%` expands differently when this example is viewed in [PreferenceSettings](#). This is because `Set` statements are not active in included topics. The including topic's `set` statements are used.

Macros defined using preference settings can take parameters. These are symbols passed in the call to the macro to define local macros that will be expanded in the output. For example,

```
* Set CONDITIONS = According to [[%BASETOPIC%]] the %WHAT% is %STATE% today (Set in ...).
```

You can call this macro passing in values for `WHAT` and `STATE`. For example:

- `%CONDITIONS{WHAT="sea" STATE="chopy"}%`
  - expands to According to [Macros](#), The sea is choppy today. (Set in [Macros](#))..

## Parameter defaults

- The special parameter name `DEFAULT` gets the value of any unnamed parameter in the macro call.
- Parameter macros can accept a `default` parameter so that they expand to something even when a value isn't passed for them in the call.

Example:

```
* Set WEATHER = It's %DEFAULT{default="raining"}%.
```

- `%WEATHER%` expands to It's raining.
- `%WEATHER{"sunny"}%` expands to It's sunny.

The standard [formatting tokens](#) can be used in parameters. They will be expanded immediately when the macro is instantiated.

**!** Note that parameters **override** all other macros, including system defined macros, in the expansion of the macro where they are used.

## Access Control Settings

These are special types of preference settings to control access to content. [AccessControl](#) explains these security settings in detail. Parameters are **not** available in access control settings.

## Local values for preferences

Certain topics (user, plugin, web, site and default preferences topics) have a problem; macros defined in those topics can have two meanings. For example, consider a user topic. A user may want to use a double-height edit box when they are editing their home topic - but **only** when editing their home topic. The rest of the time, they want to have a normal edit box. This separation is achieved using `Local` in place of `Set` in the macro definition.

For example, if the user sets the following in their home topic:

```
* Set EDITBOXHEIGHT = 10
* Local EDITBOXHEIGHT = 20
```

Then, when they are editing any other topic, they will get a 10 high edit box. However, when they are editing their home topic they will get a 20 high edit box. `Local` can be used wherever a preference needs to take a different value depending on where the current operation is being performed.

Use this powerful feature with great care! `%ALLVARIABLES%` can be used to get a listing of the values of all macros in their evaluation order, so you can see macro scope if you get confused.

## Predefined Macros

Most predefined macros return values that were either set in the configuration when Foswiki was installed, or taken from server info (such as current username, or date and time). Some, like `%SEARCH%`, are powerful and general tools.

**!** Predefined macros can be overridden by [preference settings](#) (except `TOPIC` and `WEB`)

! Plugins may extend the set of predefined macros (see individual Plugins topics for details)

💡 **Take the time to thoroughly read through ALL preference macros.** If you actively configure your site, review macros periodically. They cover a wide range of functions, and it can be easy to miss the one perfect macro for something you have in mind. For example, see `%BASETOPIC%`, `%INCLUDE%`, and the mighty `%SEARCH%`.

Your installation of Foswiki Foswiki-1.2.0-alpha, Sat, 19 May 2012, build 14862 has the following registered macros:

## ACTIVATEDPLUGINS -- list of currently activated plugins

- Syntax: `%ACTIVATEDPLUGINS%`
- Expands to: [SpreadSheetPlugin](#), [AutoViewTemplatePlugin](#), [BibtexPlugin](#), [CommentPlugin](#), [CompareRevisionsAddonPlugin](#), [EditRowPlugin](#), [FilterPlugin](#), [HistoryPlugin](#), [HomePagePlugin](#), [InterwikiPlugin](#), [jQueryPlugin](#), [MailerContribPlugin](#), [NatEditPlugin](#), [PdfPlugin](#), [PreferencesPlugin](#), [RenderListPlugin](#), [SlideShowPlugin](#), [SmiliesPlugin](#), [SubscribePlugin](#), [TablePlugin](#), [TagMePlugin](#), [TinyMCEPlugin](#), [TwistyPlugin](#), [UpdatesPlugin](#), [WysiwygPlugin](#)
- Related: [PLUGINDESCRPTIONS](#), [FAILEDPLUGINS](#), [PLUGINVERSION](#)

## ADDTOHEAD

This macro is deprecated. Please use [VarADDTOZONE](#) instead. It effectively is a shortcut for

```
%ADDTOZONE{"head" ...}%
```

## ADDTOZONE

```
%ADDTOZONE{  
  "zone"  
  ...  
}%
```

### Parameters:

- `"zone"` optional, comma-separated list of the names of zones that the content should be added to. The only zones guaranteed to exist are `head` and `script`. Defaults to `head`.
- `id` optional, `identifier` for the text being added with the `ADDTOZONE` call, to be used in the `requires` parameter of other `ADDTOZONE` calls.
  - Multiple `ADDTOZONE` calls with the same `id` parameter will simply overwrite the earlier `ADDTOZONE` call.
- `requires="..."` optional, comma separated string of `ids` of text within this `zone` that this content should follow when the zone is rendered. The content will be rendered even if a specified `id` is missing.
- `text="..."` optional, text to be added to the named zone, mutually exclusive with `topic`.
- `topic="..."` optional, full qualified `web.topic` name that contains the text to be added, mutually exclusive with `text`. Defaults to `%BASETOPIC%`
- `section="..."` optional, section of the `topic` to be added, defaults to the default section between [STARTINCLUDE](#) and [STOPINCLUDE](#).

## What is a "Zone"?

*Zones* are specific places in the output HTML that are marked by calls to the [RENDERZONE](#) macro. Zones are used to collect various content together, such as Javascript and CSS, that must be included in the output HTML in a specific order, and in a specific place.

There are two special zones called `head` and `script`. The `head` zone is rendered as part of the HTML head section. It is the catch-all container for any content supposed to be placed into the HTML head section, except Javascript, which is collected in the `script` zone.

All Javascript *must* always be added to the `script` zone exclusively, in order to grant ordering constraints among scripts are resolved properly. Never add Javascript to the `head` zone -- never add non-Javascript content to the `script` zone.

Both zones are added to the HTML head section automatically just before the closing `</head>` tag as if they were specified explicitly in the skin templates using:

```
<head>
...
%RENDERZONE{"head"}%
%RENDERZONE{"script"}%
</head>
```

You may create as many zones in addition to the standard `head` and `script` zones as you like. For any non-standard zone specified in [ADDTOZONE](#) you will also need to provide an appropriate [RENDERZONE](#).

Interesting use cases in wiki applications:

- Create a `sidebar` zone to add widgets,
- Create a `toolbar` zone to add buttons icons
- Create a `menu` zone to add menu entries

## Adding content to a zone

[ADDTOZONE](#) adds content to a zone identified with the `id` parameter. An `id` identifier is unique within the zone that they are added to. When the same `id` is used in multiple calls to [ADDTOZONE](#) the last call will win, that is previous content of the same `id` will be overwritten.

## Enforcing a linear order of content within a zone

An [ADDTOZONE](#) call may ensure that its content appears *after* the content of some other [ADDTOZONE](#) calls by specifying their `ids` in the `requires` parameter. The `requires` parameter constraints the linear order of content added to a zone. When a zone is rendered, all ordering constraints expressed via `requires` are satisfied. Those `ids` not found in a zone don't have any influence on the final ordering. Missing `ids` aren't considered an error rather than an over-specified ordering problem.


## Working with `{MergeHeadAndScriptZones}` disabled (default)

In this mode, the `head` and `script` zones are treated separately.

Even when `head` and `script` zones are treated separately, the `head` zone will always be rendered before the `script` zone, unless otherwise specified using [RENDERZONE](#) explicitly. So any content in the `script` zone that depends on content placed into the `head` zone is satisfied intrinsically as they are both rendered as specified above.

## Working with `{MergeHeadAndScriptZones}` enabled

In this mode, the `head` and `script` zones are separate when adding to them, but may be treated as merged when you call [RENDERZONE](#) if there are any dependencies specified that only exist in the opposite zone. This allows an `ADDTOZONE{"head"...}` to successfully require an `id` that has been added to `script`.

 `{MergeHeadAndScriptZones}` is provided to maintain compatibility with legacy extensions that use [ADDTOHEAD](#) to add `<script>` markup and require content that is now in the `script` zone. `{MergeHeadAndScriptZones}` will be removed from a future version of Foswiki.

---

## Example: Adding to a zone with missing dependencies

You must ensure that no `head` content (and no inline Javascript) depends on `script` content. Any such dependency will be *ignored*.

In real world application this isn't a problem as Javascript is never added to the `head` zone or Javascript zone part of the `script` zone never really depends on non-Javascript content part of the `head` zone.

HTML comment decoration which normally appears after each id's content in the rendered HTML will contain a small informative text to aid debugging.

### Example

```
%ADDTOZONE{
  "script"
  text="
<script type='text/javascript'>
  alert('test');
</script>"
  requires="some-id-that-exists-in-script"
  id="MY::TEST"
}%
```

### Result

```
<script type='text/javascript'>
  alert('test');
</script>
<!-- MY::TEST: requires= missing ids: some-id-that-exists-in-script -->
```

## Example: Adding Javascript to a page

Make sure that all inline Javascript code in the topic (if it is allowed) is added to the page using `%ADDTOZONE{"script"...requires="library-id"}%` with the appropriate `library-id` to guarantee a correct load order. For example, jQuery code should be added as follows:

```
%JQREQUIRE("shake")%
%ADDTOZONE{
  "script"
  id="MyApp::ShakePart"
  text="
<script type='text/javascript'>
  jQuery('#something').shake(3, 10, 180);
</script>"
  requires="JQUERYPLUGIN::SHAKE"
}%
```

where `"MyApp::ShakePart"` is a unique `id` to identify the text added to `script`; and `JQUERYPLUGIN::SHAKE` signifies that the content added with that identifier should appear beforehand.

## Example: Adding CSS to a page

```
%ADDTOZONE{ "head"
  id="MyCSS"
```

```
text="
  <style type='text/css' media='all'>
    @import url ('%PUBURLPATH%/%SYSTEMWEB%/MyCSS/foo.css');
  </style>"
}%
```

See also [RENDERZONE](#), [Using ADDTOZONE](#), [Updating applications to use script zone](#)

## ALLVARIABLES -- list of currently defined macros

- Syntax: `%ALLVARIABLES%`
- Expands to: a table showing all defined [macros](#) in the current context

Deprecated 2009-04-29 in favour of [SHOWPREFERENCE](#)


## AQUA -- start aqua colored text

- `AQUA` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

**Example:**

```
%AQUA% aqua text %ENDCOLOR%
```

**Expands to:** [aqua text](#)

 `<color>` text must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

**Related:** [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)

## ATTACHURL -- full URL for attachments in the current topic

- Syntax: `%ATTACHURL%`
- Expands to: `http://foswiki.trunk/pub/System/VarATTACHURL`
- Example: If you attach a file you can refer to it as `%ATTACHURL%/image.gif`
- Related: [ATTACHURLPATH](#), [PUBURL](#), [PUBURLPATH](#), [SCRIPTURL](#), [SCRIPTURLPATH](#), [FileAttachments](#)

## ATTACHURLPATH -- path of the attachment URL of the current topic

- Syntax: `%ATTACHURLPATH%`
- Expands to: `/pub/System/VarATTACHURLPATH`
- Related: [ATTACHURL](#), [PUBURL](#), [PUBURLPATH](#), [SCRIPTURL](#), [SCRIPTURLPATH](#), [FileAttachments](#)

## AUTHREALM -- authentication realm

- String defined as `{AuthRealm}` expert option in [configure](#). This is used in certain password encodings, and in login templates as part of the login prompt.
- Syntax: `%AUTHREALM%`
- Expands to: Enter your [WikiName](#). (First name and last name, no space, no dots, capitalized, e.g. `JohnSmith?`). Cancel to register if you do not have one.
- Related: [UserAuthentication](#), [SESSIONID](#), [SESSIONVAR](#), [LOGIN](#), [LOGOUT](#), [SESSION\\_VARIABLE](#)

## BASETOPIC -- base topic where an INCLUDE started

- The name of the topic where a single or nested INCLUDE started - same as `%TOPIC%` if there is no INCLUDE
- This *is* the name of the topic requested by the user.
- Syntax: `%BASETOPIC%`
- Related: [BASEWEB](#), [INCLUDINGTOPIC](#), [INCLUDE](#), [TOPIC](#)

## BASEWEB -- base web where an INCLUDE started

- The web name where the includes started, e.g. the web of the first topic of nested includes. Same as `%WEB%` in case there is no include.
- This *is* the name of the web requested by the user.
- Syntax: `%BASEWEB%`
- Related: [BASETOPIC](#), [INCLUDINGWEB](#), [INCLUDE](#), [WEB](#)

## BB -- bullet with line break

- Line break and bullet without indentation.
- Current value: BB =  
•
- Related: [BR](#), [BULLET](#), [BB2](#), [BB3](#), [BB4](#), [CARET](#), [VBAR](#)

## BB2 -- level 2 bullet with line break

- Line break and bullet, level 2.
- Current value: BB2 =  
•
- Related: [BR](#), [BULLET](#), [BB](#), [BB3](#), [BB4](#), [VBAR](#)

## BB3 -- level 3 bullet with line break

- Line break and bullet, level 3.
- Current value: BB3 =  
•
- Related: [BR](#), [BULLET](#), [BB](#), [BB2](#), [BB4](#), [VBAR](#)

## BB4 -- level 4 bullet with line break

- Line break and bullet, level 4.
- Current value: BB4 =  
•
- Related: [BR](#), [BULLET](#), [BB](#), [BB2](#), [BB3](#), [VBAR](#)


## BLACK -- start black colored text

- `BLACK` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

### Example:

```
%BLACK% black text %ENDCOLOR%
```

**Expands to:** black text

 `%<color>%` text must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

**Related:** [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)


## BLUE -- start blue colored text

- `BLUE` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

**Example:**

```
%BLUE% blue text %ENDCOLOR%
```

**Expands to:** `blue text`

 `<color>` text must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

**Related:** [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)

## BR -- line break

- Current value: `BR =`
- Related: [BULLET](#), [BB](#), [BB2](#), [BB3](#), [BB4](#), [CARET](#), [VBAR](#)


## BROWN -- start brown colored text

- `BROWN` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

**Example:**

```
%BROWN% brown text %ENDCOLOR%
```

**Expands to:** `brown text`

 `<color>` text must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

**Related:** [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)

## BULLET -- bullet character

- Current value: `BULLET = •`
- Related: [BR](#), [BB](#), [BB2](#), [BB3](#), [BB4](#), [CARET](#), [VBAR](#)

## BUTTON{"text" ...} -- renders a nice button

- Parameters:



Parameter:	Description:	Default:
"text", value="text"	text to be put on this button	
accesskey	access key used for this button	
class	extra class: use <code>foswikiRight</code> or <code>foswikiLeft</code> to specify alignment; use <code>cyan</code> , <code>red</code> , <code>green</code> for different background colors; use <code>simple</code> for a non-3D button	
href	url of the click target	#
icon	icon to be put on the left; note, this can be any icon attached to the <code>{IconSearchPath}</code> ; see also <a href="#">VarJQICON</a>	
id	html id for this button	
onclick	javascript event triggered when clicking the button	
onmouseout	javascript event triggered when the pointer leaves the button	
onmouseover	javascript event triggered when the pointer hovers over the button	
target	topic to open when clicking on the button	
title	popup title displayed when hovering over the button	
type	type of action to be performed; available actions are <ul style="list-style-type: none"> <li>button: (default) normal click button, target specified in <code>target</code> or <code>href</code> parameter</li> <li>clear: clears all input fields in the form that contains the button</li> <li>reset: resets all input fields in a form to their initial value</li> <li>submit: submits the form that contains the button</li> <li>save: same as <code>submit</code> but takes care of extra validation steps when saving a wiki topic</li> </ul>	button

- Example:

```
%BUTTON{
  "%MAKETEXT{"Submit"}%"
  click="confirm('Are your sure?')"
}%
%BUTTON{
  "%MAKETEXT{"Cancel"}%"
  icon="cross"
  target="%WEB%.%TOPIC%"
}% %CLEAR%
```

- Expands as:



- Note: BUTTONS are floating to the left by default. Take care to add a `%CLEAR%` after the `%BUTTON{...}%` so that further content does not overlap with the button.
- Related: [jQueryButton](#)

## CALC{"formula"} -- add spreadsheet calculations to tables and outside tables

- The `%CALC{"formula"}%` macro is handled by the [SpreadSheetPlugin](#). There are around 90 formulae, such as `$ABS()`, `$EXACT()`, `$EXISTS()`, `$GET()/SET()`, `$IF()`, `$LOG()`, `$LOWER()`, `$PERCENTILE()`, `$TIME()`, `$VALUE()`.
- Syntax: `%CALC{"formula"}%`
- Examples:
  - `%CALC{"$SUM($ABOVE())"}%` returns the sum of all cells above the current cell
  - `%CALC{"$EXISTS(Web.SomeTopic)}%` returns 1 if the topic exists
  - `%CALC{"$UPPER(Collaboration)}%` returns `COLLABORATION`

- Related: [IF](#), [SpreadSheetPlugin](#)

## CARET -- caret symbol

- The CARET macro is useful in TML tables.
- Current value: CARET = ^
- Related: [BR](#), [BULLET](#), [BB](#), [BB2](#), [BB3](#), [BB4](#), [VBAR](#)

## COMMENT{ *attributes* } -- insert an edit box into the topic to easily add comments.

- A `%COMMENT%` without parameters shows a simple text box.
- The following standard attributes are recognized

Name	Description	Default
<code>type</code>	This is the name of the template to use for this comment. Comment templates are defined in a Foswiki template - see <a href="#">Customisation</a> , below. If this attribute is not defined, the type is whatever is defined by <code>COMMENTPLUGIN_DEFAULT_TYPE</code> , either in this topic or in your <a href="#">WebPreferences</a> .	<code>below</code>
<code>default</code>	Default text to put into the prompt.	
<code>target</code>	Name of the topic to add the comment to	the current topic
<code>location</code>	Regular expression specifying the comment location in the target topic. Read <i>carefully</i> the <a href="#">CommentPlugin</a> documentation!	
<code>mode</code>	For compatibility with older versions only, synonymous with <code>type</code>	
<code>nonotify</code>	Set to "on" to disable change notification for target topics	<code>off</code>
<code>noform</code>	Set to "on" to disable the automatic form that is generated around your comment prompt if you don't provide a <code>FORM</code> template. See <a href="#">CommentPluginExamples:noform</a> for an example.	<code>off</code>
<code>nopost</code>	Set to "on" to disable insertion of the posted text into the topic.	<code>off</code>
<code>remove</code>	Set to "on" to remove the comment prompt after the first time it is clicked.	<code>off</code>
<code>button</code>	Button label text	Add comment


- See [CommentPlugin](#) for more information

## COVER -- current skin cover

- `%COVER%` extends the skin search path. For instance, if `SKIN` is set to `catskin, bearskin`, and `COVER` is set to `ruskin`, the skin search path becomes `ruskin, catskin, bearskin`.
- The `COVER` setting can be overridden using the URL parameter `cover`, such as `?cover=ruskin`
- Syntax: `%COVER%`
- Expands to: `%COVER%`
- See [Skins](#) for more information

## DATE -- signature format date

- Syntax: `%DATE%`
- Expands to: `19 May 2012`
- Date format defined as `{DefaultDateFormat}` in [configure](#)

 When used in a template topic, this variable will be expanded when the template is used to create a new topic. See [TemplateTopics#TemplateTopics Vars](#) for details.

- Related: [DISPLAYTIME](#), [GMTIME{"format"}](#), [SERVERTIME](#)

## DISPLAYTIME{"format"} -- formatted display time

- Formatted time - either GMT or Local server time, depending on {DisplayTimeValues} setting in [configure](#). Same format qualifiers as `%GMTIME%`
- Syntax: `%DISPLAYTIME%` OR `%DISPLAYTIME{"format"}%`
- `%DISPLAYTIME%` The time is shown as hh:mm (24 hour clock)
  - Expands to: 19 May 2012 - 19:52
- Example:

```
%DISPLAYTIME{"$hou:$min"}%
```

expands to 19:52

- Related: [GMTIME](#), [SERVERTIME](#)

## EDITACTION -- Selects an edit template

- The `EDITACTION` [preference setting](#) lets you define the use of an editaction template instead of the standard edit. If `EDITACTION` is defined as `text`, then hide the form. If `EDITACTION` is defined as `form` hide the normal text area and only edit the form.
- Syntax:

```
* Set EDITACTION = text|form
```

- Expands to: `%EDITACTION%`

**!** When `EDITACTION` is defined as `text` or `form` the Edit and Edit Raw buttons simply add `;action=text` or `;action=form` to the URL for the edit script. If you have defined an `EDITACTION` [preference setting](#) you can still edit the topic content or the form by removing the `;action=form` or `;action=text` from the edit URL in the browser and reload.

- Related: [CommandAndCGIScripts#edit](#)

## EDITTABLE{ *attributes* } -- edit tables using edit fields and other input fields

- The `%EDITTABLE{}%` macro is handled by the [EditTablePlugin](#)
- Syntax: `%EDITTABLE{ attributes }%`
- Supported attributes:

Attribute	Comment	Default
header	Specify the header format of a new table like <code>" *Food* *Drink* "</code> . Useful to start a table with only a button	(no header)
format	The format of one column when editing the table. A cell can be a text input field, or any of these edit field types: <ul style="list-style-type: none"> <li>• Text input field (1 line):   text, &lt;size&gt;, &lt;initial value&gt;  </li> <li>• Textarea input field:   textarea, &lt;rows&gt;x&lt;columns&gt;, &lt;initial value&gt;  </li> <li>• Drop down box:</li> </ul>	"text, 16" for all cells

| select, <size>, <option 1>, <option 2>, etc\* |

\* only one item can be selected

• Radio buttons:

| radio, <size\*>, <option 1>, <option 2>, etc |

\* size indicates the number of buttons per line in edit mode

• Checkboxes:

| checkbox, <size\*>, <option 1>, <option 2>, etc |

\* size indicates the number of checkboxes per line in edit mode

• Fixed label:

| label, 0, <label text> |

• Row number:

| row, <offset> |

• Date:

| date, <size>, <initial value>, <DHTML date format> | (see [Date Field Type](#))

changerows	Rows can be added and removed if "on" Rows can be added but not removed if "add" Rows cannot be added or removed if "off"	CHANGEROWS plugin setting
quietsave	Quiet Save button is shown if "on", hidden if "off"	QUIETSAVE plugin setting
include	Other topic defining the EDITTABLE parameters. The first %EDITTABLE% in the topic is used. This is useful if you have many topics with the same table format and you want to update the format in one place. Use topic or web.topic notation.	(none)
helptopic	Topic name containing help text shown below the table when editing a table. The %STARTINCLUDE% and %STOPINCLUDE% macros can be used in the topic to specify what is shown.	(no help text)
headerislabel	Table header cells are read-only (labels) if "on"; header cells can be edited if "off" or "0"	"on"
editbutton	Set edit button text, e.g. "Edit this table"; set button image with alt text, e.g. "Edit table, %PUBURL%/ %SYSTEMWEB%/DocumentGraphics/edittopic.gif"; hide edit button at the end of the table with "hide" (Note: Button is automatically hidden if an edit button is present in a cell)	EDITBUTTON plugin setting
buttonrow	Set to top to put the edit buttons above the table.	bottom
javascriptinterface	Use javascript to directly move and delete row without page refresh. Enable with "on", disable with "off".	JAVASCRIPTINTERFACE plugin setting

**Example:**

```
%EDITTABLE{ format="| text, 20 | select, 1, one, two, three |" changerows="on" }%  
| *Name* | *Type* |  
| Foo    | two    |
```

## Produces:


Name	Type
Foo	two

**Related:** See [EditTablePlugin](#) for more details

## ENCODE{"string"} -- encodes a string

- Encode character sequences in "string", by mapping characters (or sequences of characters) to an alternative character (or sequence of characters). This macro can be used to encode strings for use in URLs, to encode to HTML entities, to protect quotes, and for as many other uses as you can imagine.
- Syntax: %ENCODE{"string"}%
- Parameters:

Parameter	Description	Default
"string"	String to encode	"" (empty string)
type="encodingname"	Use a predefined encoding (see below).	Default is 'url'. Parameter <code>type</code> not be used if <code>old</code> or <code>new</code> are given.
old="tokenlist"	Comma-separated list of tokens to replace. Tokens are normally single characters, but can also be sequences of characters. The standard <a href="#">format tokens</a> may be used in this list. Each token must be unique - you cannot list the same token twice.	May not be used with <code>type</code> ; required if <code>new</code> is used
new="tokenlist"	comma-separated list of replacement tokens. The elements in this list match 1:1 with the elements in the <code>old</code> list. Again, the standard <a href="#">format tokens</a> may be used. An empty element in the <code>new</code> list will result in the corresponding token in the <code>old</code> list being deleted from the string. If the <code>new</code> list is shorter than the <code>old</code> list it will be extended to the same length using the empty element. Tokens do not have to be unique.	May not be used with <code>type</code> ; required if <code>old</code> is used

 When using `old` and `new`, be aware that the results of applying earlier tokens are not processed again using later tokens. (see examples below)

- If `ENCODE` is called with no optional parameters (e.g. %ENCODE{"string"}%) then the default `type="url"` encoding will be used.
- Predefined encodings.
  - Unless otherwise specified, the `type` parameter encodes the following "special characters"
    - all non-printable ASCII characters below space, except newline ("`\n`") and carriage return ("`\r`")
    - HTML special characters "<", ">", "&", single quote ('') and double quote (")
    - TML special characters "%", "[", "]", "@", "\_", "\*", "=", and "|"
  - `type="entity"` Encode special characters into HTML entities, like a double quote into `&#034;`. Does **not** encode `\n` (newline).
  - `type="html"` As `type="entity"` except it also encodes `\n` (newline)

- `type="safe"` Encode just the characters `'"<>%` into HTML entities.
- `type="quotes"` Escapes double quotes with backslashes (`\`), does not change any other characters
- `type="url"` Encode special characters for use in URL parameters, like a double quote into `%22`

- Examples

```
%ENCODE{"spaced name"}%= expands to
  spaced%20name
%ENCODE{"| Blah | | More blah |" old="|,$n" new="&#124;,<br />"}% expands to
  | Blah | | More blah |
  - this encoding is useful to protect special TML characters in tables.
%ENCODE{"10xx1x01x" old="1,x,0" new="A,,B"}% expands to
  ABABA
%ENCODE{"1,2" old="$comma" new=";"}% expands to
  1;2
```

- Values for HTML input fields must be entity encoded.


Example:

```
<input type="text" name="address" value="%ENCODE{ "any text" type="entity" }%" />
```

- `ENCODE` can be used to filter user input from URL parameters and similar to help protect against cross-site scripting. The safest approach is to use `type="entity"`. This can however prevent an application from fully working. You can alternatively use `type="safe"` which encodes only the characters `'"<>%` into HTML entities. When `ENCODE` is passing a string inside another macro always use double quotes (`"`) `type="quote"`. For maximum protection against cross-site scripting you are advised to install the [Foswiki:Extensions.SafeWikiPlugin](#).
- Double quotes in strings must be escaped when passed into other macros.

Example:

```
%SEARCH{ "%ENCODE{ "string with "quotes"" type="quotes" }%" noheader="on" }%
```

 When using `old` and `new`, be aware that the results of applying earlier tokens are not processed again using later tokens. For example:

```
%ENCODE{"A" old="A,B" new="B,C"}% will result in 'B' (not 'C'),
%ENCODE{"asd" old="as,d" new="d,f"}% will yield 'df', and
%ENCODE{"A" old="A,AA" new="AA,B"}% will give 'AA' and.
%ENCODE{"asdf" old="a,asdf" new="a,2"}% will give 'asdf'
```

- Related: [URLPARAM](#)


## ENDCOLOR -- end colored text

- `ENDCOLOR` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of shortcuts.

Example:

```
%GREEN% green text %ENDCOLOR%
```

Expands to: **green text**

 `<color>` text must end with `&#x26;ENDCOLOR&#x26;`. If you want to switch from one color to another one you first need to end the active color with `&#x26;ENDCOLOR&#x26;`, e.g. write

```
&#x26;RED&#x26; some text &#x26;ENDCOLOR&#x26; &#x26;GREEN&#x26; more text &#x26;ENDCOLOR&#x26;
```

- Related: [VarAQUA](#), [VarBLACK](#), [VarBLUE](#), [VarBROWN](#), [VarGRAY](#), [VarGREEN](#), [VarLIME](#), [VarMAROON](#), [VarNAVY](#), [VarOLIVE](#), [VarORANGE](#), [VarPINK](#), [VarPURPLE](#), [VarRED](#), [VarSILVER](#), [VarTEAL](#), [VarWHITE](#), [VarYELLOW](#), [DefaultPreferences](#), [StandardColors](#)

## ENDSECTION{"name"} -- marks the end of a named section within a topic

- Syntax: `&#x26;ENDSECTION{"name"}&#x26;`
- Syntax: `&#x26;ENDSECTION{type="include"}&#x26;`
- Supported parameter:

Parameter:	Description:
"name"	Name of the section.
type="..."	Type of the section being terminated; supported types "section", "include", "expandvariables", "templateonly".

- If the `STARTSECTION` is named, the corresponding `ENDSECTION` must also be named with the same name. If the `STARTSECTION` specifies a type, then the corresponding `ENDSECTION` must also specify the same type. If the section is unnamed, `ENDSECTION` will match with the nearest unnamed `&#x26;STARTSECTION&#x26;` of the same type above it.
- Related: [STARTSECTION](#)

## ENDTAB -- ending marker for a tab of a tabpane

- This closes a previously opened `TAB`.
- no parameters
- Related: [VarTAB](#), [VarTABPANE](#), [VarENDTAB](#), [jQueryPlugin](#), [jQueryTabpane](#)

## ENDTABPANE -- ending tag for tabpane widget

- This closes a previously opened `TABPANE`.
- no parameters
- Related: [VarTAB](#), [VarENDTAB](#), [VarTABPANE](#), [jQueryPlugin](#), [jQueryTabpane](#)

## ENDTWISTY

Twisty closure, complements the opening `TWISTY` tag.

- Syntax: `&#x26;ENDTWISTY&#x26;`
- Related: [VarENDTWISTY](#), [VarENDTWISTYTOGGLE](#), [VarTWISTY](#), [VarTWISTYBUTTON](#), [VarTWISTYHIDE](#), [VarTWISTYSHOW](#), [VarTWISTYTOGGLE](#)

## ENDTWISTYTOGGLE

The Twisty closure

- Syntax: `&#x26;ENDTWISTYTOGGLE&#x26;`
  - Will end the most inner unclosed Twisty Toggle section, using the proper tag
- Example: `&#x26;ENDTWISTYTOGGLE&#x26;`
- Related: [VarENDTWISTY](#), [VarENDTWISTYTOGGLE](#), [VarTWISTY](#), [VarTWISTYBUTTON](#), [VarTWISTYHIDE](#), [VarTWISTYSHOW](#), [VarTWISTYTOGGLE](#)

## ENV{"varname"} -- inspect the value of an environment variable

- Returns the current value of the environment variable in the CGI (Common Gateway Interface) environment. This is the environment that the [CommandAndCGIScripts](#) are running in.
- **Note:** For security reasons, only those environment variables whose names match the regular expression in `{AccessibleENV}` in the **Security Settings/Miscellaneous** section of `configure` can be displayed. Any other variable will just be shown as an empty string, irrespective of its real value.
- Example: `%ENV{MOD_PERL}%` displays as: not set
- If an environment variable is undefined (as against being set to the empty string) it will be returned as `not set`.
- Related: [HTTP\\_HOST](#), [REMOTE\\_ADDR](#), [REMOTE\\_PORT](#), [REMOTE\\_USER](#)

## EXAMPLETAG -- example variable

- The `%EXAMPLETAG{}%` variable is handled by the [ExamplePlugin?](#)
- Syntax: `%EXAMPLETAG{"text" format="..."}%`
- Parameter `text="..."` - example text.
- Parameter `format="..."` - format of report.
- Example: `%EXAMPLETAG{"hello" format="| $topic: $summary |"}%`
- Related: [ExamplePlugin?](#)

## EXPAND{"expression" scope="topicctoexpandin" ...}%

Expands macros in `expression` as if they were used in the topic `topicctoexpandin`. The viewer must have VIEW access to `topicctoexpandin` for this to work. All the standard formatting macros can be used in `expression`, such as `$percent` and `$quot`.

`EXPAND` can be useful when you want to pick up the value of macros defined in another topic. For example, you might want to define a set of preferences in one topic, but pick up their value in another topic (this is very useful when building reusable applications). In this case you can write:

```
* Set MYPREFERENCE = value
```

in "SettingsTopic" and then, in "MyTopic", write:

```
%EXPAND{"$percentMYPREFERENCE$percent" scope="SettingsTopic"}%
```

Of course we can also write:

```
%EXPAND{"$percentMYPREFERENCE$percent" scope="%OTHERTOPIC%"}%
```

which lets us select which other topic to get the preference value from.

Additional parameters can be passed to the macro being expanded using the standard macro syntax in the name of the macro; for example,

```
%EXPAND{"$percentMYPREFERENCE($quotdefault$quot param=$quotvalue$quot)" scope="SettingsTopic"}%
```

Notes:

- `EXPAND` is not very efficient, and should be used sparingly.
- To expand a web preference (for example, a web access control) then set `scope="Theotherweb.%WEBPREFSTOPIC%"`

## FAILEDPLUGINS -- debugging for plugins that failed to load, and handler list



- Syntax: `%FAILEDPLUGINS%`
- Expands to: See [Plugins#FAILEDPLUGINS](#)
- Related: [PLUGINDESCRPTIONS](#), [ACTIVATEDPLUGINS](#), [PLUGINVERSION](#)

## FORMAT{"list" format="" header="" footer="" separator=""} -- format a list of objects

- Syntax: `%FORMAT{"list"}%`
- Supported parameters:

Parameter:	Description:	Default:
"one, two, three"	The list to be expanded into the format. Required. Currently only two types of list data are supported; topic names ( <code>type="topic"</code> ) and plain strings ( <code>type="string"</code> ).	""
format="..."	Format string; see <a href="#">Supported formatting tokens</a> for possible values.	""
header="..."	Text to come before the formatted output	""
footer="..."	Text to come after the formatted output	""
separator="n"	Separator between formatted elements	"\$n"
type=""	Treat input list as either <code>topic</code> or <code>string</code>	"topic"

- Examples:

```
%FORMAT{"one,two,three" type="string" format=" * $item"}%
%FORMAT{"%SKIN%"
  header="the Skin setting is evaluated in this order:"
  format=" 1 =$topic="
  footer=" 1 =default="
}%
```


- Related: [SEARCH](#)

## Supported formatting tokens

If `type="topic"` (the default) the format string can contain *any* of the topic-specific format tokens specified in [FormattedSearch](#) (`$web`, `$topic`, `$parent`, `$text`, `$locked`, `$date`, `$isodate`, `$index`, `$item`, `$rev`, `$username`, `$wikiname`, `$wikiusername`, `$createdate`, `$createusername`, `$createwikiname`, `$createwikiusername`, `$summary`, `$changes`, `$formname`, `$formfield`, `$pattern`, `$count`, `$ntopics`, `$nhits`, `$pager`). In addition, the macro supports all the standard [format tokens](#).

If `type="string"` then the comma separated list is treated as a list of strings. In this case, the format tokens `$index` and `$item` will return the position of the item in the list (1-based), and the item itself, respectively. Note that a comma can be embedded in the data using the standard formatting token `$comma`.

The FORMAT macro is currently only of use in formatting lists of topics, or of simple strings. It will be extended in future releases to add the capability to render other object types.

 For more sophisticated handling of string lists, consider installing `Foswiki:Extensions.FilterPlugin`.

- Related: [SEARCH](#), [ENCODE](#), [FormattedSearch](#)

## FORMFIELD{"fieldname"} -- renders a field in the form attached to some topic

- Syntax: `%FORMFIELD{"fieldname"}%`
- Supported parameters:

Parameter:	Description:	Default:
"fieldname"	The name of a <a href="#">Data form</a> field	required
topic="..."	Topic where form data is located. May be of the form <code>Web.TopicName</code>	Current topic
format="..."	Format string. <code>\$value</code> expands to the field value, and <code>\$name</code> expands to the field name, <code>\$title</code> to the field title, <code>\$form</code> to the name of the form the field is in. The <a href="#">standard format tokens</a> are also expanded.	"\$value"
default="..."	Text shown if the field is defined in the topic, but the field value is empty. For example, a text field for which all the content has been deleted.	""
alttext="..."	Text shown if the field is not defined in the topic (even if it is specified in the form definition). For example, this is used when a field exists in the form definition, but the referring topic hasn't been edited since it was added.	""
rev="n"	Specify a revision of the topic. If not specified, defaults to the most recent rev (or the viewed rev if viewing an old rev of the same topic)	

- Example:

```
%FORMFIELD{"ProjectName"
  topic="Projects.SushiProject"
  default="(no project name given)"
  alttext="ProjectName field not found in form"
}%
```

- Related: [SEARCH](#)

## GENPDF -- Insert a link that generated a PDF for the current or identified topic

- Syntax: `%GENPDF%`
- Expands to: `%GENPDF%`
- Example: To link to an alternate topic, write `%GENPDF{"AnotherTopic"}%` or write `%GENPDF{"AnotherTopic" link="pdf version"}%` to override the link text.
- Related: Installed as part of [GenPDFAddOn](#)

## GMTIME{"format"} -- formatted GM time

- Syntax: `%GMTIME%` OR `%GMTIME{"format"}%`
- `%GMTIME%` uses the default date format defined by the `{DefaultDateFormat}` setting in [configure](#)
  - expands to 19 May 2012 - 19:53
- Supported special format tokens:

Token:	Unit:	Example
<code>\$seconds</code>	seconds	59
<code>\$minutes</code>	minutes	59
<code>\$hours</code>	hours	23
<code>\$day</code>	day of month	31
<code>\$wday</code>	day of the Week (Sun, Mon, Tue, Wed, Thu, Fri, Sat)	Thu
<code>\$dow</code>	day of the week (Sun = 0)	2
<code>\$week</code>	number of week in year (ISO 8601)	34
<code>\$month</code>	short name of month	Dec
<code>\$mo</code>	2 digit month	12
<code>\$year</code>	4 digit year	1999


\$ye	2 digit year	99
\$tz	either "GMT" (if set to gmtime), or "Local" (if set to servertime)	GMT
\$iso	ISO format timestamp	2012-05-19T19:53:00Z
\$rcs	RCS format timestamp	2012/05/19 19:53:00
\$http	E-mail & http format timestamp	Sat, 19 May 2012 19:53:00 GMT
\$epoch	Number of seconds since 00:00 on 1st January, 1970	1337457180

- Tokens can be shortened to 3 characters
- Example:

```
%GMTIME{"$day $month, $year - $hour:$min:$sec"}%
```

expands to

```
19 May, 2012 - 19:53:00
```

 When used in a template topic, this macro will be expanded when the template is used to create a new topic. See [TemplateTopics#TemplateTopics Vars](#) for details.

- Related: [GMTIME](#), [REVINFO](#), [SERVERTIME](#)


## GRAY -- start gray colored text

- GRAY is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

**Example:**

```
%GRAY% gray text %ENDCOLOR%
```

**Expands to:** gray text

 `%<color>%` text must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

**Related:** [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)


## GREEN -- start green colored text

- GREEN is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

**Example:**

```
%GREEN% green text %ENDCOLOR%
```

**Expands to:** green text

 `%<color>%` text must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

**Related:** [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)

## GROUPINFO{"name"} -- retrieve details about a group

- **Syntax:** `%GROUPINFO%`
  - Expands to: comma-separated list of all groups
- **Syntax:** `%GROUPINFO{"groupname"}%`
  - Expands to: comma-separated list of users in that group
- **Parameters:**

<code>format</code>	Format of a single user or group in the list. <ul style="list-style-type: none"><li>◦ <code>\$name</code> expands to the group name, and (for users list only)</li><li>◦ <code>\$wikiname</code>, <code>\$username</code> and <code>\$wikiusername</code> to the relevant strings.</li><li>◦ <code>\$allowschange</code> returns 0 (false) or 1 (true) if that group can be modified by the current user.</li><li>◦ <code>\$allowschange(UserWikiName)</code> returns 0 (false) or 1 (true) if that group can be modified by the specified user (does not work for groups yet.).</li><li>◦ The standard <a href="#">FormatTokens</a> are also supported.</li></ul>	<code>\$name</code> for groups list, <code>\$wikiusername</code> for users list
<code>separator</code>	separator between items in the list	<code>,</code>
<code>header</code>	Header for the list	<code>"</code>
<code>footer</code>	Footer for the list	<code>"</code>
<code>zeroreresults</code>	If set, and there are no Groups or Members that can be shown, the <code>header</code> and <code>footer</code> are suppressed, and this text is output	<code>undefined</code>
<code>show</code>	filter the output list of Groups - can be set to <code>all</code> , <code>allowschange</code> , <code>denychange</code> , <code>allowschange(UserWikiName)</code> , <code>denychange(UserWikiName)</code>	<code>all</code>
<code>expand</code>	Set false if users should not be expanded from nested groups. Default behavior is to expand all nested groups into a flat list of users.	<code>1</code>
<code>limit</code>	If set, limits the number of results to this	<code>infinity</code>
<code>limited</code>	If limit is set, and the list is truncated, this text will be added at the end of the list	<code>"</code>

**Note:** GROUPINFO will not list members that are hidden from the current authenticated user. If the current user does not have VIEW authority for a user's topic, then the user will not be shown as a group member.


- Related: [USERINFO](#)

## GROUPS -- a formatted list of groups

**Deprecated - do not use. Use [VarGROUPINFO](#) instead**

- Expands to a formatted list of user groups in your Foswiki.
- **Syntax:** `%GROUPS%`
- The macro is intended to be used in [WikiGroups](#), to allow a group listing for various user mapping managers.
- Related: [REMOTE\\_USER](#), [USERINFO](#), [USERNAME](#), [WIKIUSERNAME](#), [WIKIUSERSTOPIC](#)

## H -- help icon

- Current value: H = 
- Related: I, [ICON](#), M, N, P, Q, S, T, U, X, Y

## HISTORY{ *attributes* } -- control attributes of tables and sorting of table columns

- The `%HISTORY{%}` macro is handled by the [HistoryPlugin](#)
- Syntax: `%HISTORY{ attributes }%`

Argument	Description	Default value
none	Default layout: a simple list of topic revisions using the default format (see below)	
"format" or format="string"	Format of one line, may include any variable which is supported by macro <a href="#">REVINFO</a>	"r\$rev - \$date - \$wikiusername"
topic="topic"	Topic name, can be in <code>web.topic</code> format	current topic
web="web"	Web name	current web
versions="number or range"	Number or range (format: <code>from..to</code> ). Examples: To get version 2, write: <code>versions="2"</code> To get version 2 to 3, write: <code>versions="2..3"</code> To get version 2 to the latest, write: <code>versions="2.."</code> To get all versions up to version 5, write: <code>versions="..5"</code> To get all versions up to but not including the latest, write: <code>versions="..-1"</code> To get the versions from 1 to 5 in reverse order, write: <code>versions="5..1"</code>	all versions in the order latest to first
header="text"	Text to print before the list. May contain the tokens <code>\$next</code> and <code>\$previous</code> which will be evaluated if there are newer or older revisions available for the topic that are not listed according to <code>versions</code> (or <code>rev1</code> , <code>rev2</code> , <code>nrev</code> ). These tokens take the syntax <code>\$next{'some text' url='url'}</code> (the same for <code>\$previous</code> ). 'some text' is the text which should be printed, 'url' is the url for the corresponding link. The tokens <code>\$rev1</code> , <code>\$rev2</code> , <code>\$nrev</code> in 'text' or 'url' will be replaced by appropriate values for the next or previous block of revisions. See the attached <code>oopshistory.tpl</code> for an example of how to use this.	"\$next"
footer="text"	Text to print after the list. May contain the tokens <code>\$next</code> and <code>\$previous</code> (see header)	"\$previous"

Deprecated (but supported) parameters:

Argument	Description	Default value
nrev="number"	Number of revisions to show. Ignored if <code>versions</code> is specified, or if both <code>rev1</code> and <code>rev2</code> are specified.	10
rev2="number"	Newest revision to show	<code>rev1+nrev</code> if <code>rev1</code> is specified, latest revision otherwise
rev1="number"	Oldest revision to show	<code>rev2-nrev</code>
reverse="boolean"	Show newest revisions first, if on	"on"

## Additional macros

The following macros are replaced only if there is a corresponding `%HISTORY%` on the page. If more than one `%HISTORY%` is used on the same page, the values from the last one will be used.

- `%HISTORY_REV1%`: Oldest revision from the printed history
- `%HISTORY_REV2%`: Latest revision from the printed history
- `%HISTORY_NREV%`: Number of the printed revisions
- `%HISTORY_MAXREV%`: Latest available revision of the topic


## HOMETOPIC -- home topic in each web

- Syntax: `%HOMETOPIC%`
- Expands to: `WebHome`, renders as [WebHome](#)
- Related: [NOTIFYTOPIC](#), [STATISTICSTOPIC](#), [TOPIC](#)

## HTTP -- get HTTP headers

- Called with the name of an HTTP header field, returns its value. Capitalization and the use of hyphens versus underscores are not significant.
- Syntax: `%HTTP%`
- Syntax: `%HTTP{"Header-name"}%`
- Examples:

<code>%HTTP%</code>	
<code>%HTTP{"Accept-language"}%</code>	en,*
<code>%HTTP{"User-Agent"}%</code>	Mozilla/5.0 (Macintosh; U; Intel Mac OS X; C -) AppleWebKit?/532.4 (KHTML, like Gecko) Qt/4.6.1 Safari/532.4

 You can see the HTTP headers your browser sends to the server on a number of sites e.g. <http://www.ericiguere.com/tools/http-header-viewer.html>

- Related: [HTTPS](#), [REMOTE\\_ADDR](#), [REMOTE\\_PORT](#), [REMOTE\\_USER](#)


## HTTP\_HOST -- environment variable

- Syntax: `%HTTP_HOST%`
- Expands to: `foswiki.trunk`
- Related: [ENV](#), [REMOTE\\_ADDR](#), [REMOTE\\_PORT](#), [REMOTE\\_USER](#)

























## HTTPS -- get HTTPS headers

- The same as `%HTTP%` but operates on the HTTPS environment variables present when the SSL protocol is in effect. Can be used to determine whether SSL is turned on.
- Syntax: `%HTTPS%`
- Syntax: `%HTTPS{"Header-name"}%`
- Related: [HTTP](#), [REMOTE\\_ADDR](#), [REMOTE\\_PORT](#), [REMOTE\\_USER](#)

## I -- idea icon

- Current value: I = 
- Related: [H](#), [ICON](#), [M](#), [N](#), [P](#), [Q](#), [S](#), [T](#), [U](#), [X](#), [Y](#)

`ICON{"name" alt="" default="name"}` -- small documentation graphic or icon of common attachment types

- Generates a small graphic image from the set attached to [DocumentGraphics](#). Images typically have a 16x16 pixel size. You can select a specific image by name, or you can give a full filename, in which case the type of the file will be used to select one of a collection of common file type icons.
- if you specify an icon which cannot be found, the one specified in the `default` parameter will be used (and if that fails, the 'else' icon will be used)
- Syntax: `=%ICON{ "filename or icon name" [ default="filename or icon name" ] [ alt="alt text to be added to the HTML img tag" ] }%`
- Examples:
  - `%ICON{"flag-gray"}%` displays as 
  - `%ICON{"pdf"}%` displays as 
  - `%ICON{"docx" default="doc"}%` displays as 
  - `%ICON{"smile.pdf"}%` displays as 
  - `%ICON{"/dont/you/dare/smile.pdf"}%` returns 
  - `%ICON{"data.unknown" alt="Unknown file type"}%` displays as 
  - `%ICON{"data.unknown"}%` displays as 
  - `%ICON{"http://trunk.foswiki.org/pub/System/DocumentGraphics/xsl.gif"}%` displays 
- Graphic samples:  arrowbright,  bubble,  choice-yes,  hand
- File type samples:  bmp,  doc,  gif,  hlp,  html,  mp3,  pdf,  ppt,  txt,  xls,  xml,  zip
- Related: [ICONURL](#), [ICONURLPATH](#), [DefaultPreferences](#), [FileAttachments](#), [DocumentGraphics](#)

## ICONURL{"name" default="name"} -- URL of small documentation graphic or icon

- Generates the full URL of a [DocumentGraphics](#) image, which Foswiki renders as an image. The related `%ICON{"name"}%` generates the full HTML img tag. Specify image name or full filename (see [ICON](#) for details on filenames.)
- Syntax: `%ICONURL{"name"}%`
- Examples:
  - `%ICONURL{"arrowbright"}%` returns   
`http://foswiki.trunk/pub/System/DocumentGraphics/arrowbright.png`
  - `%ICONURL{"novel.pdf"}%` returns `http://foswiki.trunk/pub/System/DocumentGraphics/pdf.png`
  - `%ICONURL{"/queen/boheme.mp3"}%` returns   
`http://foswiki.trunk/pub/System/DocumentGraphics/mp3.png`
- Related: [ICONURLPATH](#), [ICON](#), [DefaultPreferences](#), [FileAttachments](#), [DocumentGraphics](#)

## ICONURLPATH{"name" default="name"} -- URL path of small documentation graphic or icon

- Generates the relative URL path of a [DocumentGraphics](#) image, typically used in an HTML img tag. Specify image name or full filename (see [ICON](#) for details on filenames.)
- Syntax: `%ICONURLPATH{"name"}%`
- Examples:
  - `%ICONURLPATH{"locktopic"}%` returns `/pub/System/DocumentGraphics/locktopic.png`
  - `%ICONURLPATH{"eggysmell.xml"}%` returns `/pub/System/DocumentGraphics/xml.png`
  - `%ICONURLPATH{"/doc/xhtml.xsl"}%` returns `/pub/System/DocumentGraphics/xsl.png`
- Related: [ICONURL](#), [ICON](#), [DefaultPreferences](#), [FileAttachments](#), [DocumentGraphics](#)

## IF{"condition" ...} -- simple conditionals

- Evaluate a condition and show one text or another based on the result. See details in [IfStatements](#).
- Syntax:

```
%IF{"CONDITION" then="THEN" else="ELSE"}%
```

shows

"THEN" if "CONDITION" evaluates to TRUE, otherwise "ELSE" will be shown

- Example:

```
%IF{"defined FUNFACTOR"
  then="FUNFACTOR is defined"
  else="FUNFACTOR is not defined"
}%
```

renders as

```
FUNFACTOR is not defined
```

- Related: [\\$\(F\)](#) of [SpreadSheetPlugin](#)

## INCLUDE{"topic"} -- include other topic.

- Syntax: `%INCLUDE{"topic" ...}%` (See also the URL form of `%INCLUDE%`)
- Supported parameters:

Parameter:	Description:	Default:
"SomeTopic"	The name of a topic located in the current web, i.e. <code>%INCLUDE{"WebNotify"}%</code>	
"Web.Topic"	A topic in another web, i.e. <code>%INCLUDE{"System.SiteMap"}%</code>	
"Web.Topic, <a href="#">SomeOtherTopic?</a> , <a href="#">OrOtherTopic?</a> "	A list of topics - <code>INCLUDE</code> will include the first topic that exists and the user has permission to VIEW. If a <code>section</code> is also specified, it will use the first topic that has that section defined in it.	
pattern="..."	Include a subset of a topic or a web page. Specify a <a href="#">RegularExpression</a> that contains the text you want to keep in parenthesis, e.g. <code>pattern="(from here.*?to here)"</code> . <a href="#">IncludeTopicsAndWebPages</a> has more.	none
rev="2"	Include a previous topic revision; N/A for URLs	top revision
warn="off"	Warn if topic include fails: Fail silently (if <code>off</code> ); output default warning (if set to <code>on</code> ); else, output specific text (use <code>\$topic</code> for topic name)	<code>%INCLUDEWARNING%</code>
headingoffset="1"	Adds the given offset to any HTML headings generated in the included text. Works on headings defined by HTML tags as well as headings defined using foswiki markup.	0
section="name"	Includes only the specified named section, as defined in the included topic by the <code>STARTSECTION</code> and <code>ENDSECTION</code> macros. Nothing is shown if the named section does not exist. <code>section=""</code> is equivalent to not specifying a section	
PARONE="val 1" PARTWO="val 2"	Any other parameter will be defined as a macro within the scope of the included topic. The example parameters on the left will result in <code>%PARONE%</code> and <code>%PARTWO%</code> being defined within the included topic.	


- Examples: See [IncludeTopicsAndWebPages](#)
- Related: [BASETOPIC](#), [BASEWEB](#), [INCLUDE\("URL"\)](#), [INCLUDE\("doc:"\)](#), [INCLUDINGTOPIC](#), [INCLUDINGWEB](#), [STARTINCLUDE](#), [STOPINCLUDE](#), [STARTSECTION](#), [ENDSECTION](#)


## INCLUDE{"url"} -- include a web page



- **Syntax:** `%INCLUDE{"http://..." ...}%` (See also the topic form of `%INCLUDE%`)
- **Supported parameters:**

Parameter:	Description:	Default:
<code>"http://..."</code>	<p>A full qualified URL, i.e.</p> <pre>%INCLUDE{"http://foswiki.org:80/index.html"}%</pre> <p>Supported content types are <code>text/html</code> and <code>text/plain</code>.</p> <div style="background-color: #e0ffe0; padding: 5px; border: 1px solid #c0ffc0;"> <p> If the URL resolves to an attachment file on the server this will <i>automatically</i> translate to a server-side include.</p> </div>	
<code>pattern="..."</code>	<p>Include a subset of a topic or a web page. Specify a <a href="#">RegularExpression</a> that contains the text you want to keep in parenthesis, e.g. <code>pattern="(from here.*?to here)"</code>. <a href="#">IncludeTopicsAndWebPages</a> has more.</p>	none
<code>raw="on"</code>	<p>When a page is included, normally Foswiki will process it, doing the following: 1) Alter relative links to point back to originating host, 2) Remove some basic HTML tags (<code>html</code>, <code>head</code>, <code>body</code>, <code>script</code>) and finally 3) Remove newlines from HTML tags spanning multiple lines. If you prefer to include <i>exactly</i> what is in the source of the originating page set this to <code>on</code>.</p> <p><code>raw="on"</code> is short for <code>disableremoveheaders="on"</code>, <code>disableremovescript="on"</code>, <code>disableremovebody="on"</code>, <code>disablecompresstags="on"</code> and <code>disablerewriteurls="on"</code>.</p>	disabled
<code>literal="on"</code>	<p>While using the <code>raw</code> option will indeed include the raw content, the included content will still be processed and rendered like regular topic content. To disable parsing of the included content, set the <code>literal</code> option to <code>"on"</code>.</p>	disabled
<code>disableremoveheaders="on"</code>	Bypass stripping headers from included HTML (everything until first <code>&lt;/head&gt;</code> tag)	disabled
<code>disableremovescript="on"</code>	Bypass stripping all <code>&lt;script&gt;</code> tags from included HTML	disabled
<code>disableremovebody="on"</code>	Bypass stripping the <code>&lt;/body&gt;</code> tag and everything around over and below it	disabled
<code>disablecompresstags="on"</code>	Bypass replacing newlines in HTML tags with spaces. This compression step rewrites unmatched <code>&lt;'s</code> into <code>&amp;lt;</code> entities unless bypassed	disabled
<code>disablerewriteurls="on"</code>	Bypass rewriting relative URLs into absolute ones	disabled
<code>warn="off"</code>	Warn if URL include fails: Fail silently (if <code>off</code> ); output default warning (if set to <code>on</code> ); else, output specific text (use <code>\$topic</code> for topic name) appended with the http error information.	<code>%INCLUDEWARNING%</code> <a href="#">preferences</a> setting

 JavaScript in included webpages is filtered out as a security precaution per default (disable filter with `disableremovescript` parameter)

 Foswiki by default is configured to deny URL format includes.

- Examples: See [IncludeTopicsAndWebPages](#)
- Related: [INCLUDE\("topic"\)](#)

## INCLUDE{"doc:"} -- include Foswiki embedded module documentation


- Syntax: `%INCLUDE{"doc:Foswiki::Func" ...}%`
- Supported parameters:

Parameter:	Description:	Default:
"doc:..."	A full qualified Foswiki module, i.e. <code>%INCLUDE{"doc:Foswiki::Func"%}</code> . The module must be found on the Foswiki lib path	
level="1"	Override the root heading level to the specified number	
pattern="..."	Include a subset of the module. Specify a <a href="#">RegularExpression</a> that contains the text you want to keep in parenthesis, e.g. <code>pattern="(from here.*?to here)"</code> . <a href="#">IncludeTopicsAndWebPages</a> has more.	none

- Examples: See [System/PerlDoc?module=Foswiki::Func](#)
- Related: [INCLUDE\("topic"\)](#) [INCLUDE\("URL"\)](#)

## INCLUDINGTOPIC -- name of topic that includes current topic


- The name of the topic that includes the current topic - same as `%TOPIC%` in case there is no include
- If a topic is used in a chain of `INCLUDEs`, `INCLUDINGTOPIC` is set to the topic directly `INCLUDING` this one, *NOT* the topic that has been requested by the user (which is set in `BASETOPIC`)

 While this Macro may appear to work, unless you require the subtle difference between `INCLUDINGTOPIC` and `BASETOPIC`, you probably should use `BASETOPIC`

- Syntax: `%INCLUDINGTOPIC%`
- Related: [BASETOPIC](#), [INCLUDINGWEB](#), [INCLUDE](#), [TOPIC](#)

## INCLUDINGWEB -- web that includes current topic

- The web name of the topic that includes the current topic - same as `%WEB%` if there is no `INCLUDE`.
- If a topic is used in a chain of `INCLUDEs`, `INCLUDINGWEB` is set to the topic directly `INCLUDING` this one, *NOT* the web that has been requested by the user (which is set in `BASEWEB`)

 While this Macro may appear to work, unless you require the subtle difference between `INCLUDINGWEB` and `BASEWEB`, you probably should use `BASEWEB`

- Syntax: `%INCLUDINGWEB%`
- Related: [BASEWEB](#), [INCLUDINGTOPIC](#), [INCLUDE](#), [WEB](#)

## JQICON{"name" ...} -- render an image

- This renders an icon image as found on an *icon search path*. The icon search path is [configured](#) in `{JQueryPlugin}{IconSearchPath}` and defaults to [FamFamFamSilkIcons](#), [FamFamFamSilkCompanion1Icons](#), [FamFamFamFlagIcons](#), [FamFamFamMinIcons](#), [FamFamFamMintIcons](#)!. The named icon will be picked found first on this path of topics where icons are attached to. The `JQICON` leverages the general icon loading mechanism as implemented by the `JQueryPlugin` and used by [BUTTON](#) as well.
- Parameters:

Parameter:	Description:	Default:
------------	--------------	----------

"name"	name of the icon to display	
class	additional css class for the img tag	
alt	alt attribute	
title	title attribute	
format	format string used to render the icon; known variables to be used in the format string are: <ul style="list-style-type: none"> <li>◦ \$iconPath: url path</li> <li>◦ \$iconClass: css class as specified by the class parameter</li> <li>◦ \$iconAlt: alt attribute-value; if the alt parameter to JQICON is set, this expands to alt='...'</li> <li>◦ \$iconTitle: title attribute-value; if the title parameter to JQICON is set, this expands to title='...'</li> </ul>	<pre>&lt;img src='\$iconPath' class='\$iconClass' \$iconAlt\$iconTitle/&gt;</pre>

- Example:

```
%JQICON{"tick" alt="alternative content" title="this is a tick icon"}%
%JQICON{"cross"}%
%JQICON{"disk"}%
%JQICON{"star"}%
%JQICON{"lightbulb"}%
%JQICON{"camera"}%
%JQICON{"date"}%
```

Produces:       

- Related: [VarJQICONPATH](#), [VarICON](#), [jQueryPlugin](#), [FamFamFamSilkIcons](#)

## JQICONPATH{"name"} -- render the urlpath to an image

- This is a shortcut for

```
%JQICON{"name" format="$iconPath"}%
```

- Example: `/pub/System/FamFamFamSilkIcons/tick.png`
- Related: [VarJQICON](#), [VarICONURL](#), [jQueryPlugin](#)

## JQPLUGINS{"plugins" ... } -- display a summary of available plugins

- Parameters:

Parameter:	Description:	Default:
"plugins"	this is a regular expression that the plugin identifier must match to be displayed	
format	format string to render information for each matching plugin; known variables to be used in the format string are: <ul style="list-style-type: none"> <li>◦ active: state of the plugin: displays <b>(active)</b> when this plugin is loaded on the current page</li> <li>◦ author: author of the plugin</li> <li>◦ documentation: plugin documentation topic defaults to <code>%SYSTEMWEB%.jQuery\$name</code></li> <li>◦ homepage: link to the homepage of this</li> </ul>	<pre>1 &lt;a href="\$homepage"&gt;\$name&lt;/a&gt; \$active \$version \$author</pre>

	<ul style="list-style-type: none"> <li>third party plugin</li> <li>index: the current index in the list of all plugins being displayed</li> <li>name: name of the plugin as can be used in <a href="#">JQREQUIRE</a></li> <li>summary: short description what this plugin does; most plugins provide this piece of information in the <code>summary</code> section of the documentation topic</li> <li>tags: list of TML macros this plugin implements</li> <li>version: version of the plugin as provided by the author of this plugin</li> </ul>
header	header string prepended to the output; empty when no plugin matches
footer	footer string appended to the output; empty when no plugin matches
separator	separator put between each plugin rendered in a row <span style="float: right;">\$n</span>
tagformat	format string to render a link to any tag documentation a plugin implements <span style="float: right;">[[%SYSTEMWEB%.Var\$tag][\$tag]]</span>

- Example:

```
%JQPLUGINS{
  "treeview|slimbox"
  header=" * JQuery Plugins:$n"
  format=" * [[$documentation][$name]] v$version was developed by [[$homepage][$author]]"
}%
```

Produces:

- JQuery Plugins:
  - [Slimbox](#) v2.04 was developed by [Christophe Beyls](#)
  - [Treeview](#) v1.4 was developed by [Joern Zaefferer](#)
- Related: [JQueryPlugin](#)

## JQREQUIRE{"plugin, plugin, ... "} -- enable a plugin on the current page

- This macro will load a list of plugins to be added to the current page. Use [JQPLUGINS](#) to display the list of available and active plugins. While loading a plugin, additional plugins it may depend on are loaded as well. Information about these dependencies is stored within the plugins themselves and can't be changed. Dependencies also make sure the javascript code is added to the html page in the right order. It uses [ADDDTOZONE](#) to aggregate javascript and css at the right place on the html page.
- Parameters:

Parameter:	Description:	Default:
"plugin[, plugin, plugin]"	list of plugins to be loaded	
warn	(on/off) allows you to switch off warnings when a plugin was not found	on

- Related: [JQueryPlugin](#), [VarJQPLUGINS](#), [VarADDDTOZONE](#)

## JQTHEME{"name" ...} -- switch jQuery UI theme

- Foswiki's default UI theme is configured in `$Foswiki::cfg{JQueryPlugin}{JQueryTheme}` and defaults to `base`. Use [configure](#) to change this site wide. Use `JQTHEME` if you decide to use a different theme on

the current page.

- Note: some Foswiki skins may come with their own jQuery UI matching the overall user experience of the web design.
- Parameters:

Parameter:	Description:	Default:
"name"	name of theme: JQueryPlugin knows the following themes <code>base</code> , <code>lightness</code> , <code>redmod</code> , <code>smoothness</code> ; additional themes maybe created using the <a href="#">themeroller</a> and installed to <code>/pub/System/JQueryPlugin/\$name</code>	base

- Related: [jQueryUI](#)

## LANG -- the lang attribute of generated HTML pages

- In templates the lang attribute is defined like this:

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="%LANG%" lang="%LANG%">
```

- The value is read from [configure](#) setting `{Site}{Locale}`
- Do not confuse LANG with LANGUAGE
- Syntax: `%LANG%`
- Expands to: `en`

## LANGUAGE -- current user's language

- Returns the language code for the language used as the current user. This is the language actually used by Foswiki (e.g. in user interface).
- The language is detected from the user's browser, unless some site/web/user/session-defined setting overrides it:
  - If the LANGUAGE preference is set, it's used as user's language instead of any language detected from the browser.
  - Avoid defining LANGUAGE at a non per-user way, so each user can choose his/her preferred language.
- Related: [LANGUAGES](#)

## LANGUAGES -- list available languages

- List the languages available (as PO files). These are the languages in which the user interface is available.
- Syntax: `%LANGUAGES{...}%`
- Supported parameters:

Parameter:	Description:	Default:
<code>format</code>	format for each item. See below for format tokens available in the format string.	" * \$langname"
<code>separator</code>	separator between items.	"\n" (newline)
<code>marker="selected"</code>	Text for \$marker if the item matches selection	"selected"
<code>selection="%LANGUAGE%"</code>	Current language to be selected in list	(none)

- `format` tokens:

Token	Meaning
<code>\$langname</code>	language's name, as informed by the translators
<code>\$langtag</code>	language's tag. Ex: <code>en</code> , <code>pt-br</code> , etc.

- Example: `<select>%LANGUAGES{format="<option $marker value='$langtag'>$langname</option>" selection="%LANGUAGE%"}%</select>` creates an option list of the available languages with the current language selected


## LIME -- start lime colored text

- `LIME` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

**Example:**

```
%LIME% lime text %ENDCOLOR%
```

**Expands to:** lime text

 `%<color>%` text must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

**Related:** [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)


## LOCALSITEPREFS -- web.topicname of site preferences topic

- The full name of the local site preferences topic. These local site preferences overload the system level preferences defined in [System.DefaultPreferences](#).
- Syntax: `%LOCALSITEPREFS%`
- Expands to: `Main.SitePreferences`, renders as [SitePreferences](#)

## LOGIN -- present a full login link


- Syntax: `%LOGIN%`
- Expand to: [Log In](#)
- Related: [UserAuthentication](#), [SESSIONID](#), [SESSIONVAR](#), [LOGOUT](#), [SESSION\\_VARIABLE](#)

## LOGOUT -- present a full logout link


 You are already logged out, so `%LOGOUT` expands to an empty string

- Syntax: `%LOGOUT%`
- Expand to:
- Related: [UserAuthentication](#), [SESSIONID](#), [SESSIONVAR](#), [LOGIN](#), [SESSION\\_VARIABLE](#)

## M -- moved to... icon

- Current value: M = 
- Related: [H](#), [I](#), [ICON](#), [N](#), [P](#), [Q](#), [S](#), [T](#), [U](#), [X](#), [Y](#)

## MAINWEB -- synonym for USERSWEB

 **Deprecated.** Please use `%USERSWEB%` instead.

## MAKETEXT -- creates text using Foswiki's I18N infrastructure

Strings captured in the `MAKETEXT` macro are automatically mapped to the current user's selected language via `locale/*.po` translation files.

- **Syntax:** `%MAKETEXT{"string" args="..."}%`
- **Supported parameters:**

Parameter	Description	Default
"text" or string="text"	The text to be displayed (the <i>translatable string</i> ).	<i>none</i>
args="param1, param2"	a comma-separated list of arguments to be interpolated in the string, replacing <code>[_N]</code> placeholders in it.	<i>none</i>

- **Examples:**

```
%MAKETEXT{string="Notes:"}%
expands to:
Notes:

%MAKETEXT{
  "If you have any questions, please contact [_1]."
  args="%WIKIWEBMASTER%"
}%
expands to:
If you have any questions, please contact arthur@visiblearea.com.

%MAKETEXT{
  "Did you want to [[[[_1]][reset [_2]'s password]]]?"
  args="%SYSTEMWEB%.ResetPassword,%WIKIUSERNAME%"
}%
expands to:
Did you want to [[System.ResetPassword][reset Main.WikiGuest's password]]?
```

- **Notes:**
  - An ampersand (&) followed by one ascii alphabetic character (a...z, A...Z) in the translatable string will be expanded to an access key string. For example, `&X` will expand to `<span class='foswikiAccessKey'>X</span>`. If you want to write an actual ampersand, either follow it with a non-alphabetic character or write two consecutive ampersands (&&).
  - Translatable strings starting with underscores (`_`) are reserved. You **cannot** use translatable phrases starting with an underscore.
  - Make sure that the translatable string is *constant*. Do **not** include `%MACROS%` inside the translatable strings as they will be expanded **before** the `%MAKETEXT{...}%` itself is handled. You can, however, use macros in the `args`, as shown in the examples above.
  - The string will be output in English if no mapping can be found in the `.po` translation file for the current user's selected language.


## MAROON -- start maroon colored text

- `MAROON` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

### Example:

```
%MAROON% maroon text %ENDCOLOR%
```

**Expands to:** maroon text

 `%<color>%` text must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

Related: [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)


## META -- displays meta-data

Provided mainly for use in templates, this macro generates the parts of the topic view that relate to meta-data (attachments, forms etc.).

Syntax: `%META{ "item" ...}%`


Parameters:

- "form" - generates the table showing the form fields. See [Form Definition](#).
- "attachments" - generate a table of attachments
  - `all="on"` to show hidden attachments.
  - `title="..."` to show a title - only if attachments are displayed.
  - `template="..."` to use a custom template for the rendering of attachments; default `attachtables` is used.
- "moved" - if a topic was moved or renamed, generates a message with details and a revert link
  - `prefix="..."`: Prefix that goes before the moved message, but only if the message is generated, default "".
  - `suffix="..."`: Prefix that goes after the moved message, but only if the message is generated, default "".
- "parent" - display details of ancestor topics
  - `dontrecurse="on"`: By default recurses up tree, this has some cost. Equivalent to `depth=1`
  - `depth="..."`: Return only the specified ancestor.
  - `nowebhome="on"`: Suppress WebHome.
  - `format="..."`: Format string used to display each parent topic where `$web` expands to the web name, and `$topic` expands to the topic name; default: `"[[$web.$topic]]$topic]"`
  - `separator="..."`: Separator between parents; default " > "
  - `prefix="..."`: Prefix that goes before parents, but only if there are parents, default "".
  - `suffix="..."`: Suffix, only appears if there are parents; default "".
- "formfield" - display the value of a single form field
  - `name="..."`: name of the field.
  - `newline="..."`: by default, each newline character will be rewritten to `<br />` to allow metadata that contains newlines to be used in tables, etc. `$n` indicates a newline character.
  - `bar="..."`: by default, each vertical bar is rewritten to an HTML entity so as to not be mistaken for a table separator.

 Use of "formfield" is **deprecated** in favour of the much more powerful `QUERY` macro.

Related: [QUERY](#)

## METASEARCH -- special search of meta data

 `METASEARCH` is **deprecated** in favour of the new and much more powerful query type search. See [SEARCH](#) and [QuerySearch](#).

- Syntax: `%METASEARCH{...}%`
- Supported parameters:

Parameter:	Description:	Default:
<code>type="topicmoved"</code>	What sort of search is required?	Required



	<p>"<code>topicmoved</code>" if search for a topic that may have been moved</p> <p>"<code>parent</code>" if searching for topics that have a specific parent i.e. its children</p> <p>"<code>field</code>" if searching for topics that have a particular form field value (use the <code>name</code> and <code>value</code> parameters to specify which field to search).</p>	
<code>web="%WEB%"</code>	Wiki web to search: A web, a list of webs separated by whitespace, or all webs.	Current web
<code>topic="%TOPIC%"</code>	The topic the search relates to, for <code>topicmoved</code> and <code>parent</code> searches	All topics in a web
<code>name</code>	form field to search, for <code>field</code> type searches. May be a regular expression (see <a href="#">SEARCH</a> ).	
<code>value</code>	form field value, for <code>field</code> type searches. May be a regular expression (see <a href="#">SEARCH</a> ).	
<code>title="Title"</code>	Text that is prefixed to any search results	empty
<code>format="..."</code>	Custom format results. Supports same format strings as <a href="#">SEARCH</a> . See <a href="#">FormattedSearch</a> for usage & examples	Results in table
<code>default="none"</code>	Default text shown if no search hit	Empty

- **Examples:**

```
%METASEARCH{
  type="topicmoved"
  web="%WEB%"
  topic="%TOPIC%"
  title="This topic used to exist and was moved to: "
}%
```

You may want to use this in [WebTopicViewTemplate?](#) and [WebTopicNonWikiTemplate?](#):

```
%METASEARCH{
  type="parent"
  web="%WEB%"
  topic="%TOPIC%"
  title="Children: "
}%

%METASEARCH{
  type="field"
  name="Country"
  value="China"
}%
```

- Related: [SEARCH](#), [META](#)

## N -- "new" icon

- Current value: N = NEW
- Related: [H](#), [I](#), [ICON](#), [M](#), [P](#), [Q](#), [S](#), [T](#), [U](#), [X](#), [Y](#)

## NAVY -- start navy colored text

- `NAVY` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

### Example:

```
%NAVY% navy text %ENDCOLOR%
```

Expands to: navy text

**!** %<color>% text must end with %ENDCOLOR%. If you want to switch from one color to another one you first need to end the active color with %ENDCOLOR%, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

Related: [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)

## NOP -- template text not to be expanded in instantiated topics

- Syntax: %NOP%
  - In normal topic text, expands to <nop>, which prevents expansion of adjacent macros and wikiwords
  - When the topic containing this is used as a template for another topic, it is removed.
- Syntax: %NOP{...}% **deprecated**
  - In normal topic text, expands to whatever is in the curly braces (if anything).

**!** This is deprecated. Do not use it. Use %STARTSECTION{type="templateonly"}% .. %ENDSECTION{type="templateonly"}% instead (see [TemplateTopics](#) for more details).

- Related: [STARTSECTION](#), [TemplateTopics](#)

## NOTIFYTOPIC -- name of the notify topic

- Syntax: %NOTIFYTOPIC%
- Expands to: [WebNotify](#), renders as [WebNotify](#)
- Related: [HOMETOPIC](#), [STATISTICSTOPIC](#), [TOPIC](#)

## OLIVE -- start olive colored text

- `OLIVE` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

### Example:

```
%OLIVE% olive text %ENDCOLOR%
```

Expands to: olive text

**!** %<color>% text must end with %ENDCOLOR%. If you want to switch from one color to another one you first need to end the active color with %ENDCOLOR%, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

Related: [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)


## ORANGE -- start orange colored text

- **ORANGE** is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

**Example:**

```
%ORANGE% orange text %ENDCOLOR%
```


**Expands to:** orange text

 **%<color>% text must end with %ENDCOLOR%. If you want to switch from one color to another one you first need to end the active color with %ENDCOLOR%, e.g. write**

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

**Related:** [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)

## P -- pencil icon

- Current value: P = 
- Related: [H](#), [I](#), [ICON](#), [M](#), [N](#), [Q](#), [S](#), [T](#), [U](#), [X](#), [Y](#)


## PINK -- start pink colored text

- **PINK** is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

**Example:**

```
%PINK% pink text %ENDCOLOR%
```

**Expands to:** pink text

 **%<color>% text must end with %ENDCOLOR%. If you want to switch from one color to another one you first need to end the active color with %ENDCOLOR%, e.g. write**

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

**Related:** [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)

## PLUGINDESCRIPTIONS -- list of plugin descriptions

- **Syntax:** %PLUGINDESCRIPTIONS%
- **Expands to:**
  - [SpreadSheetPlugin](#) (1.1.6, 13989): Add spreadsheet calculations like "\$SUM(\$ABOVE())" to Foswiki tables and other topic text
  - [AutoViewTemplatePlugin](#) (2010-12-10, 10257): Automatically sets VIEW\_TEMPLATE and EDIT\_TEMPLATE
  - [BibtexPlugin](#) (2.2.0, \$Rev: 14850 (2012-05-18) \$): Embed BibTeX entries.
  - [CommentPlugin](#) (2.0.3, 13766): Quickly post comments to a page without an edit/save cycle
  - [CompareRevisionsAddonPlugin](#) (1.1.6, 13320):
  - [EditRowPlugin](#) (3.0.1, 13852): Inline edit for tables
  - [FilterPlugin](#) (2.08, 13591): Substitute and extract information from content by using regular expressions
  - [HistoryPlugin](#) (1.9, 15950): Shows a complete history of a topic
  - [HomePagePlugin](#) (\$Date: 2008-12-15 04:49:56 +1100 (Mon, 15 Dec 2008) \$, 1340): Allow User specified

- home pages - on login
- [InterwikiPlugin](#) (1.1.2, 12956): Link ExternalSite:Page text to external sites based on aliases defined in a rules topic
- [jQueryPlugin](#) (4.4.0, \$Rev\$): jQuery JavaScript library for Foswiki
- [MailerContribPlugin](#) (9 Jul 2010, 5752): Supports e-mail notification of changes
- [NatEditPlugin](#) (6.03, 14035): A Wikiwyg Editor
- [PdfPlugin](#) (0.1.0, 14686): Generate PDF files from pages.
- [PreferencesPlugin](#) (1.1.2, 13319): Allows editing of preferences using fields predefined in a form
- [RenderListPlugin](#) (2.2.5, 13720): Render bullet lists in a variety of formats
- [SlideShowPlugin](#) (3.0.0, 12411): Create web based presentations based on topics with headings
- [SmiliesPlugin](#) (05 Dec 2011, 13319): Render smilies like 😊 as icons
- [SubscribePlugin](#) (03 Dec 2008, 13286): This is a companion plugin to the [MailerContrib](#). It allows you to trivially add a "Subscribe me" link to topics to get subscribed to changes.
- [TablePlugin](#) (1.140, 14095): Control attributes of tables and sorting of table columns
- [TagMePlugin](#) (2.0.1, 14819): Tag wiki content collectively to find content by keywords
- [TinyMCEPlugin](#) (1.2.5, 13499): Integration of the Tiny MCE WYSIWYG Editor
- [TwistyPlugin](#) (1.6.15, 13319): Twisty section Javascript library to open/close content dynamically
- [UpdatesPlugin](#) (0.30, \$Rev\$): Checks Foswiki.org for updates
- [WysiwygPlugin](#) (1.1.3, 13286): Translator framework for WYSIWYG editors
- Related: [ACTIVATEDPLUGINS](#), [FAILEDPLUGINS](#), [PLUGINVERSION](#)

## PLUGINVERSION -- the version of a Foswiki Plugin, or the Foswiki Plugins API

- Syntax: `%PLUGINVERSION{"name"}%` to get the version of a specific plugin
- Example: `%PLUGINVERSION{"InterwikiPlugin"}%` expands to `$Rev: 12956 $`
- Syntax: `%PLUGINVERSION%` to get the version of the API
- Expands to: 2.3
- Related: [WIKIVERSION](#), [ACTIVATEDPLUGINS](#), [FAILEDPLUGINS](#), [PLUGINDESCRIPTIONS](#)

## POPUPWINDOW{"topic" ...} -- opens a topic or url in a new window

- Parameters:

Parameter:	Description:	Default:
"topic", topic="topic", topic="web.topic"	Topic to open	
url	URL to open (if topic is not used)	
label	Link label	the topic or the url
template	View template to call when viewing a topic; not used for URLs	"viewplain"
width	Width of window	"600"
height	Height of window	"480"
toolbar	Show toolbars?	"0"
scrollbars	Show scrollbars?	"1"
status	Show status?	"1"
location	Show location bar?	"0"
resizable	Is the window resizable?	"1"
left	Left position	"0"
top	Top position	"0"
center	Center the window?	"0"
menubar	Show menubar?	"0"

```
createnew
```

Create a new window for each popup?

```
"1"
```

- Example with topic link:

```
%POPUPWINDOW{"Macros" label="Open this topic in a new window"}%
```

Generates: [Open this topic in a new window](#)

- Example with url:

```
%POPUPWINDOW{url="http://foswiki.org"}%
```

Generates: <http://foswiki.org>

- Enable `POPUPWINDOW` by writing `%JQREQUIRE{"popupwindow"}%` on the page

## PUBURL -- the base URL of attachments

- Syntax: `%PUBURL%`
- Expands to: `http://foswiki.trunk/pub`
- Example: You can refer to a file attached to another topic with `%PUBURL%/%WEB%/OtherTopic/image.gif`
- Related: [ATTACHURL](#), [ATTACHURLPATH](#), [PUBURLPATH](#), [SCRIPTURL](#), [SCRIPTURLPATH](#), [FileAttachments](#)

## PUBURLPATH -- the base URL path of attachments

- Syntax: `%PUBURLPATH%`
- Expands to: `/pub`
- Related: [ATTACHURL](#), [ATTACHURLPATH](#), [PUBURL](#), [SCRIPTURL](#), [SCRIPTURLPATH](#), [FileAttachments](#)


## PURPLE -- start purple colored text

- `PURPLE` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

**Example:**

```
%PURPLE% purple text %ENDCOLOR%
```


**Expands to:** purple text

 `<color>` text must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

**Related:** [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)

## Q -- question icon

- Current value: Q = 
- Related: [H](#), [I](#), [ICON](#), [M](#), [N](#), [P](#), [S](#), [T](#), [U](#), [X](#), [Y](#)

## QUERY -- get the value of meta-data

- Uses the query syntax described in [QuerySearch](#) to get information about meta-data from one specified topic.
  - supports formatted access to formfields and other meta-data in topics using the same syntax as

- is used in `IF` and `SEARCH` statements,
- gives access to **all** meta-data, including that added by extensions,
- supports reporting values using JSON and other standards, simplifying the retrieval of meta-data for REST applications,
- replaces the `FORMFIELD` macro for most applications.
- **Syntax:** `%QUERY( "query" )%`
- See `QuerySearch` for more details of how to write queries
- **Parameters:**
  - `style="stylename"` - set the output format (see below)
  - `rev="version"` - operate on the given version of the current topic. Note that this will *only* affect simple queries that refer to the current topic, such as `form.name`. More complex queries that use searches or indirection to refer to other topics always use the **latest** version of those topics.
- **Examples:**

```

Get the name of the form in the current topic:
%QUERY{"form.name"}%

Get the value of the 'Firstname' form field in
the current topic:
%QUERY{"fields[name='Firstname'].value"}%

Get the value of the 'Firstname' form field in
the current topic (shorthand version):
%QUERY{"Firstname"}%

Get a list of all the names of attachments on
the topic 'System.DocumentGraphics':
%QUERY{"'System.DocumentGraphics'/attachments.name"}%

Get configuration setting {NameFilter}:
%QUERY{"{NameFilter}"}%

```

Plain strings (such as field values) are returned *without* quotes. Simple arrays of scalars are also returned without quotes, in a comma-separated list (beware of values that contain commas!).

More complex data structures (e.g. arrays of hashes) will only be returned if `style="perl"` or `style="json"` are set - else will return a string containing 'undef'.

You can make the macro generate different output formats using the `style` parameter:

- `style="perl"` - generates values as Perl code strings generated by running through [CPAN:Data::Dumper](#)
- `style="json"` - generates values as JSON strings, suitable for reading by browsers.

Only some configuration settings are available via `QUERY`: `{ScriptSuffix}`, `{LoginManager}`, `{AuthScripts}`, `{LoginNameFilterIn}`, `{AdminUserLogin}`, `{AdminUserWikiName}`, `{SuperAdminGroup}`, `{UsersTopicName}`, `{AuthRealm}`, `{MinPasswordLength}`, `{Register}{AllowLoginName}`, `{Register}{EnableNewUserRegistration}`, `{Register}{NeedVerification}`, `{Register}{RegistrationAgentWikiName}`, `{AllowInlineScript}`, `{DenyDotDotInclude}`, `{UploadFilter}`, `{NameFilter}`, `{AccessibleCFG}`, `{AntiSpam}{EmailPadding}`, `{AntiSpam}{EntityEncode}`, `{AntiSpam}{HideUserDetails}`, `{AntiSpam}{RobotsAreWelcome}`, `{Stats}{TopViews}`, `{Stats}{TopContrib}`, `{Stats}{TopicName}`, `{UserInterfaceInternationalisation}`, `{UseLocale}`, `{Site}{Locale}`, `{Site}{CharSet}`, `{DisplayTimeValues}`, `{DefaultDateFormat}`, `{Site}{LocaleRegexes}`, `{UpperNational}`, `{LowerNational}`, `{PluralToSingular}`, `{EnableHierarchicalWebs}`, `{WebMasterEmail}`, `{WebMasterName}`, `{NotifyTopicName}`, `{SystemWebName}`, `{TrashWebName}`, `{SitePrefsTopicName}`, `{LocalSitePreferences}`, `{HomeTopicName}`, `{WebPrefsTopicName}`, `{UsersWebName}`, `{TemplatePath}`,

```
{LinkProtocolPattern}, {NumberOfRevisions}, {MaxRevisionsInADiff},
{ReplaceIfEditedAgainWithin}, {LeaseLength}, {LeaseLengthLessForceful},
{Plugins}{WebSearchPath}, {PluginsOrder}, {Cache}{Enabled}, {Validation}{Method},
{Register}{DisablePasswordConfirmation}
```



- Related: [META QuerySearch](#)

## QUERYPARAMS -- show paramaters to the query

- Expands the parameters to the query that was used to display the page.
- Syntax: `%QUERYPARAMS{...}%`
- Supported parameters:

Parameter:	Description:	Default:
<code>format="..."</code>	Format string for each entry	<code>\$name=\$value</code>
<code>separator="..."</code>	Separator string	<code>separator="\$n"</code> (newline)
<code>encoding="entity"</code> <code>encoding="safe"</code> <code>encoding="html"</code> <code>encoding="quotes"</code> <code>encoding="url"</code>	Control how special characters are encoded. If this parameter is not given, "safe" encoding is performed which HTML entity encodes the characters ' <code>"&lt;&gt;</code> '. <hr/> <b>entity:</b> Encode special characters into HTML entities, like a double quote into <code>&amp;#034;</code> . Does <b>not</b> encode <code>\n</code> or <code>\r</code> . <hr/> <b>safe:</b> Encode characters ' <code>"&lt;&gt;</code> ' into HTML entities. (this is the default) <hr/> <b>html:</b> As <code>type="entity"</code> except it also encodes <code>\n</code> and <code>\r</code> <hr/> <b>quotes:</b> Escape double quotes with backslashes ( <code>\</code> ), does not change other characters <hr/> <b>url:</b> Encode special characters for URL parameter use, like a double quote into <code>%22</code>	<code>type="safe"</code>

- The following escape sequences are expanded in the format string:

Sequence:	Expands To:
<code>\$name</code>	Name of the parameter
<code>\$value</code>	String value of the parameter. Multi-valued parameters will have a "row" for each value.
<code>\$n</code> or <code>\$n()</code>	New line. Use <code>\$n()</code> if followed by alphanumeric character, e.g. write <code>Foo\$n()Bar</code> instead of <code>Foo\$nBar</code>
<p> Most macros accept parameter strings which are split over multiple lines. This is usually more readable than using <code>\$n</code> tokens. If you are familiar with <a href="#">sectional includes</a>, you might also consider <a href="#">nested sectional includes</a> to hold the newline content outside of the parameter string entirely.</p>	
<p> Note that newline is <i>not</i> a line break. The browser will wrap the lines together. If you require a line break, displaying the results on two lines, use <code>%BR%</code>. Or use two consecutive newlines to create a TML "Paragraph".</p>	
<code>\$nop</code> or <code>\$nop()</code>	Is a "no operation". This token gets removed; useful for nested search
<code>\$quot</code>	Double quote ( <code>"</code> ) ( <code>\</code> also works)

<code>\$percent</code>	Percent sign (%) ( <code>\$percent</code> also works)
<code>\$dollar</code>	Dollar sign (\$)
<code>\$lt</code>	Less than sign (<)
<code>\$gt</code>	Greater than sign (>)
<code>\$amp</code>	Ampersand (&)
<code>\$comma</code>	Comma (,)

- Example:

```
%QUERYPARAMS {
    format="<input type='hidden' name='$name' value='$value' encoding="entity" />"
}%
```


### Security warning!

Using QUERYPARAMS can easily be misused for cross-site scripting unless specific characters are entity encoded. By default QUERYPARAMS encodes the characters ' "<>% into HTML entities (same as encoding="safe") which is relatively safe. The safest is to use encoding="entity". When passing QUERYPARAMS inside another macro always use double quotes (") combined with using QUERYPARAMS with encoding="quote". For maximum security against cross-site scripting you are advised to install the Foswiki:Extensions.SafeWikiPlugin.

- See also [QUERYSTRING](#), [URLPARAM](#)

## QUERYSTRING -- full, unprocessed string of parameters to this URL

- String of all the URL parameters that were on the URL used to get to the current page. For example, if you add `?name=Samantha;age=24;eyes=blue` to this URL you can see this in action. This string can be appended to a URL to pass parameter values on to another page.

 URLs built this way are typically restricted in length, typically to 2048 characters. If you need more space than this, you will need to use an HTML form and `=%QUERYPARAMS%=`

- Syntax: `%QUERYSTRING%`
- Expands to: `cover=print;viewtemplate=plain`
- Related: [QUERYPARAMS](#), [URLPARAM](#)


## RED -- start red colored text

- RED is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

**Example:**

```
%RED% red text %ENDCOLOR%
```

**Expands to:** red text

 `<color>% text` must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

**Related:** [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)



## REMOTE\_ADDR -- environment variable

- Syntax: `%REMOTE_ADDR%`
- Expands to: `127.0.0.1`
- Related: [ENV](#), [HTTP\\_HOST](#), [REMOTE\\_PORT](#), [REMOTE\\_USER](#)

## REMOTE\_PORT -- environment variable

- Syntax: `%REMOTE_PORT%`
- Expands to:
- Related: [ENV](#), [HTTP\\_HOST](#), [REMOTE\\_ADDR](#), [REMOTE\\_USER](#)

## REMOTE\_USER -- environment variable

- Syntax: `%REMOTE_USER%`
- Expands to:
- Related: [ENV](#), [HTTP\\_HOST](#), [REMOTE\\_ADDR](#), [REMOTE\\_PORT](#), [USERNAME](#), [WIKINAME](#), [WIKIUSERNAME](#)

## RENDERLIST -- render bullet lists in a variety of formats

- The `%RENDERLIST%` macro is handled by the [RenderListPlugin](#)
- Syntax: `%RENDERLIST%`
- Example:

```
%RENDERLIST{"org" focus="Sales.WestCoastTeam"}%  
  * [[Eng.WebHome][Engineering]]  
    * [[Eng.TechPubs][Tech Pubs]]  
  * [[Sales.WestCoastTeam][Sales]]  
    * [[Sales.EastCoastTeam][East Coast]]  
    * [[Sales.WestCoastTeam][West Coast]]
```

Expands as:



- Related: [RenderListPlugin](#)

## RENDERZONE

```
%RENDERZONE{"zone" ...}%
```

See [ADDTOZONE](#) for an explanation of *zones*.

Parameters:

- "zone" required, name of the zone.
- `format="..."` optional, format string for each item added to the zone, default:

```
$item <!--<literal> $id $missing</literal>-->
```

Tokens:

- `$id` - id of the [ADDTOZONE](#) call within the `zone` currently being rendered.
- `$item` - text of the [ADDTOZONE](#) call within the `zone` currently being rendered.

- `$zone` - the "zone" currently being rendered.
- `$missing` - if the `ADDTOZONE` call being rendered required any `id` which was not found, then `$missing` is the `missingtoken` parameter; empty string otherwise.
- `$missingids` - comma separated list of ids that were required by the `ADDTOZONE` call currently being rendered but weren't found within this `zone`.
- `missingtoken="..."` optional, this will be the string assigned to the `$missing` format token for use in the `format` parameter. Default:


```
$id: requires= missing ids: $missingids
```

- `chomp="on"` remove leading and trailing whitespace from formatted items, can be useful for pretty-printing and compression.
- `header="..."` optional, prepended to the output
- `footer="..."` optional, appended to the output
- `separator="..."` optional, put between each item of a zone

Supports the [standard format tokens](#) in all parameters.

#### Notes:

- `header` and `footer` are **not** output if there is no content in the zone (nothing has been `ADDTOZONEd`). However they **are** output if the output is the empty string (at least one `ADDTOZONE` has been processed).
- Zones are cleared after being rendered; they are only ever rendered once.
- `head` and `script` are *automatic zones*. They don't require a corresponding `RENDERZONE` anywhere in the templates - they are automatically inserted before the `</head>` tag in the output HTML page.
- Normally, dependencies between individual `ADDTOZONE` statements are resolved within each zone. However, if `{MergeHeadAndScriptZones}` is enabled in [configure](#), then `head` content which requires an `id` that only exists in `script` will be re-ordered to satisfy this dependency.

 `{MergeHeadAndScriptZones}` will be removed from a future version of Foswiki.

See also [ADDTOZONE](#) for more information on zones.

## REVARG -- `&rev=n` URL revision parameter of current topic

- Syntax: `%REVARG%`
- `%REVARG%` If a topic revision is requested in the URL, it returns the revision of the current topic suitable for concatenation to the view query parameters. Otherwise returns an empty string.
- Expands to: *(simulated)* `&rev=3` *(actual)*
- Related: [REVINFO](#), [REVTITLE](#)

## REVINFO -- revision information of current topic

- Syntax: `%REVINFO%`
- `%REVINFO%` is equivalent to `%REVINFO{format="r1.$rev - $date - $wikiusername"}%`
  - Expands to: `r1 - 26 Jun 2011 - 21:40:14 - ProjectContributor`
- Related: [REVARG](#), [REVTITLE](#)

## REVINFO{"format"} -- formatted revision information of topic

- Syntax: `%REVINFO{"format"}%`
- `%REVINFO%` is equivalent to `%REVINFO{format="r1.$rev - $date - $wikiusername"}%`
  - Expands to: `r1 - 26 Jun 2011 - 21:40:14 - ProjectContributor`
- Supported parameters:

Parameter:	Description:	Default:
"format"	Format of revision information, see supported formatting tokens below	"r\$rev - \$date - \$wikiusername"
web="..."	Name of web	Current web
topic="..."	Topic name	Current topic
rev="5"	Specific revision number	Latest revision

- Supported formatting tokens:

Token:	Unit:	Example
\$web	Name of web	Current web
\$topic	Topic name	Current topic
\$rev	Revision number.	5
\$username	Login username of revision	jsmith
\$wikiusername	<a href="#">WikiName</a> of revision	JohnSmith
\$wikiname	WikiName with Main web prefix	Main.JohnSmith
\$date	Revision date. Actual date format defined as {DefaultDateFormat} in <a href="#">configure</a>	21 Sep 2006
\$time	Revision time	23:24:25
\$iso	Revision date in ISO date format	2006-09- 22T06:24:25Z
\$min, \$sec, etc.	Same date format qualifiers as <a href="#">GMTIME{"format"}</a>	

- Example, returns revision info of revision 43:

```
%REVINFO{"$date - $wikiusername" rev="43"}%
```

- To get the latest revision, even when looking at an older revision:


```
%REVINFO{"$rev" rev="-1"}%
```

- Related: [REVARG](#), [REVTITLE](#)

## REVTITLE -- (r1) The requested revision as displayed in topic breadcrumbs

- Syntax: `%REVTITLE%`
- `%REVTITLE%` If a topic revision is requested in the URL, it returns the printable revision of the current topic revision. Otherwise returns an empty string.
- Expands to: *(simulated)* (r3) *(actual)*
- Related: [REVINFO](#), [REVARG](#)

## S -- red star icon

- Current value: S = 
- Related: [H](#), [I](#), [ICON](#), [M](#), [N](#), [P](#), [Q](#), [T](#), [U](#), [X](#), [Y](#)

## SCRIPTNAME -- name of current script


- The name of the current script is shown, including script suffix, if any (for example `viewauth.cgi`)
- Syntax: `%SCRIPTNAME%`
- Expands to: `view`
- Related: [SCRIPTSUFFIX](#), [SCRIPTURL](#), [SCRIPTURLPATH](#)


## SCRIPTSUFFIX -- script suffix

- Some Foswiki installations require a file extension for CGI scripts, such as `.pl` or `.cgi`
- Syntax: `%SCRIPTSUFFIX%`
- Expands to:
- Related: [SCRIPTNAME](#), [SCRIPTURL](#), [SCRIPTURLPATH](#)

## SCRIPTURL{"script"} -- URL of script

- Syntax: `%SCRIPTURL%` OR `%SCRIPTURL{"script"}%`
- `%SCRIPTURL%` returns the base URL of scripts - expands to <http://foswiki.trunk/bin>
- Expands to: `http://foswiki.trunk/bin/script`
- Example: To get the authenticated version of the current topic you can write `%SCRIPTURL{"viewauth"}%/%WEB%/%TOPIC%` which expands to `http://foswiki.trunk/bin/viewauth/System/Macros`


 In most cases you should use `%SCRIPTURLPATH{"script"}%` instead, as it works with URL rewriting much better

 The `edit` script should always be used in conjunction with `?t=%GMTIME{"$epoch"}%` to ensure pages about to be edited are not cached in the browser

- Related: [PUBURL](#), [SCRIPTNAME](#), [SCRIPTSUFFIX](#), [SCRIPTURLPATH](#)

## SCRIPTURLPATH{"script"} -- URL path of script

- As `%SCRIPTURL{"script"}%`, but doesn't include the protocol and host part of the URL
- Syntax: `%SCRIPTURL%` OR `%SCRIPTURLPATH{"script"}%`
- Expands to: `/bin/script`

 The `edit` script should always be used in conjunction with `?t=%GMTIME{"$epoch"}%` to ensure pages about to be edited are not cached in the browser

- Related: [PUBURLPATH](#), [SCRIPTNAME](#), [SCRIPTSUFFIX](#), [SCRIPTURL](#)

## SEARCH{"text"} -- search content

- Inline search, shows a search result embedded in a topic
- Syntax: `%SEARCH{"text" ...}%`
- Supported parameters:

Parameter:	Description:	Default:
<code>"text"</code>	Search term. Is a keyword search, literal search, regular expression search, or query, depending on the <code>type</code> parameter. <a href="#">SearchHelp</a> has more	required
<code>search="text"</code>	(Alternative to above)	N/A
<code>web="Name"</code> <code>web="Main, Know"</code> <code>web="all"</code>	Comma-separated list of webs to search. You can specifically <b>exclude</b> webs from an <code>all</code> search using a minus sign - for example, <code>web="all, -Secretweb"</code> . The special word <code>all</code> means all webs that do <b>not</b> have the <code>NOSEARCHALL</code> preference set to on in their <a href="#">WebPreferences</a> . Note that <a href="#">AccessControls</a> are respected when searching webs; it is <b>much</b> better to use them than <code>NOSEARCHALL</code> . Wildcards are not currently supported	Current web

	for web names.	
topic="WebPreferences" topic="*Bug"	Limit search to topics: A topic, a topic with asterisk wildcards, or a list of topics separated by comma. <b>Note</b> this is a list of <b>topic names</b> and must <b>not</b> include web names.	All topics in a web
excludetopic="Web*" excludetopic="WebHome, WebChanges"	Exclude topics from search: A topic, a topic with asterisk wildcards, or a list of topics separated by comma. <b>Note</b> this is a list of <b>topic names</b> and must <b>not</b> include web names.	None
scope="topic" scope="text" scope="all"	Search topic name (title); the text (body) of topic; or all (title and body)	"text"
type="keyword" type="word" type="literal" type="regex" type="query"	Control how the search is performed when scope="text" or scope="all" <hr/> keyword: use Google-like controls as in soap "web service" -shampoo; searches word parts: using the example, topics with "soapsuds" will be found as well, but topics with "shampoos" will be excluded <hr/> word: identical to keyword but searches whole words: topics with "soapsuds" will not be found, and topics with "shampoos" will not be excluded <hr/> literal: search for the exact string, like web service <hr/> regex: use a <a href="#">RegularExpression</a> search like soap;web service;!shampoo; to search on whole words use \bsoap\b <hr/> query: <a href="#">query search</a> of form fields and other meta-data, like (Firstname='Emma' OR Firstname='John') AND Lastname='Peel'	%SEARCHVAR- DEFAULTTYPE% <a href="#">preferences</a> setting (literal)
order="topic" order="created" order="modified" order="editby" order="formfield(name) "	Sort the results of search by the topic names, topic creation time, last modified time, last editor's <a href="#">WikiName</a> , or named field of <a href="#">DataForms</a> . The sorting is done web by web; if you want to sort across webs, create a <a href="#">formatted</a> table and sort it with <a href="#">TablePlugin</a> 's initsort. Note that dates are sorted most recent date <b>last</b> (i.e at the bottom of the table).	Sort by topic name
limit="all" limit="16"	Limit the number of topics from which results will be returned. This is done after sorting if order is specified. Note that this does not limit the number of hits from the same topic when you have multiple="on".	All results
date="..."	limits the results to those pages with latest edit time in the given <a href="#">time interval</a> .	All results
reverse="on"	Reverse the direction of the search	Ascending search
casesensitive="on"	Case sensitive search	Ignore case
bookview="on"	<a href="#">BookView</a> search, e.g. show complete topic text. Very resource demanding. Use only with small result sets	Show entire topic content.
nonoise="on"	Shorthand for nosummary="on" nosearch="on" nototal="on" zeroresults="off"	Off

	<code>noheader="on" noempty="on"</code>	
<code>nosummary="on"</code>	Show topic title only	Show topic summary
<code>nosearch="on"</code>	Suppress search string	Show search string
<code>noheader="on"</code>	Suppress default search header <b>Topics: Changed: By:</b> , unless a <code>header</code> is explicitly specified	Show default search header, unless search is inline and a format is specified (Cairo compatibility)
<code>nototal="on"</code>	Do not show number of topics found	Show number
<code>zeroreresults="off"</code> or <code>zeroreresults="..."</code>	Suppress/replace <i>all</i> output if there are no hits (the boolean nature of the setting uses <code>true</code> , <code>false</code> , <code>on</code> , <code>off</code> , <code>0</code> so those cannot be used as a format string on their own (insert a <code>\</code> to escape them))- can also be set to a <b>FormattedSearch</b> string to customise the output	<code>zeroreresults="on"</code> - displays the summary, and number of topics found. "Number of topics: 0"
<code>noempty="on"</code>	Suppress results for webs that have no hits.	Show webs with no hits
<code>header="..."</code> <code>format="..."</code> <code>footer="..."</code>	Custom format results: see <b>FormattedSearch</b> for usage & examples	Results in table
<code>expandvariables="on"</code>	Expand embedded macros before applying a <b>FormattedSearch</b> on a search hit. Useful to show the expanded text, e.g. to show the result of a <b>SpreadSheetPlugin</b> <code>%CALC{}</code> instead of the formula	Raw text
<code>multiple="on"</code>	Multiple hits per topic. Each hit can be <b>formatted</b> . The last token is used in case of a regular expression <code>;</code> and search	Only one hit per topic
<code>nofinalnewline="on"</code>	If <code>on</code> , the search variable does not end in a line by itself. Any text continuing immediately after the SEARCH macro on the same line will be rendered as part of the table generated by the search, if appropriate. This feature is only active when format is defined.	<code>on</code>
<code>recurse="on"</code>	Recurse into subwebs, if subwebs are enabled. Note: recurse will currently search subwebs of explicitly excluded webs. ( <code>web="all, -Sandbox" recurse="on"</code> ) will still search subwebs of <code>Sandbox</code> . This behavior is likely to change in a future release.	<code>off</code>
<code>separator=", "</code>	Line separator <i>between</i> search hits (only used when <code>format=</code> is set) uses <b>FormatTokens</b> . If <code>separator</code> is not defined, the default is <code>"\$n"</code> (newline). Not defining the separator will additionally cause a newline to be added after a header and before a footer.	<code>"\$n"</code> (Newline)
<code>headingoffset="1"</code>	Adds the given offset to any HTML headings generated in the search result. Works on headings defined by HTML tags as well as headings defined using foswiki markup.	0
<code>newline="%BR%"</code>	Line separator <i>within</i> a search hit. Useful if you want to put multi-line content into a table cell, for example	<code>"\$n"</code> (Newline)

	if the <code>format=""</code> parameter contains a <code>\$pattern()</code> that captures more than one line.	
<code>pagesize="25"</code>	number of items to show per page	"25"
<code>showpage="1"</code>	Page of items to show (starts at 1) (over-ridden by the value specified by the URL parameter hash from <code>\$previousurl</code> and <code>\$nexturl</code> )	"1"
<code>pager="on"</code>	appends the pager to the footer format (the quickest way to add paging to your SEARCHes is to just add <code>pager="on"</code> ) <i>Note:</i> the default pager (when <code>pagerformat</code> is not defined) requires the parameters to the SEARCH to not change while paging, as it uses <code>\$previousurl</code> and <code>\$nexturl</code> which use a hash of the Macro's parameters to override the value of <code>showpage</code> . If you use time variable parameters, you will need to define your own <code>pagerformat</code> .	"off"
<code>pagerformat="text"</code>	Custom format results: see <a href="#">FormattedSearch</a> for usage & examples	filled from skin template
<code>groupby="none"</code>	<i>Warning: this option is liable to change dramatically (and potentially incompatibly) in the next major release of foswiki.</i> Setting to "none" applies only to multi-web SEARCHs, and means the <code>header</code> and <code>footer</code> are only output <i>once</i> - at the beginning and end of the list of results, and the <code>order</code> parameter is applied over the entire set of results (this setting removes the legacy that results are always partitioned by web) see <a href="#">SiteChanges</a> for an example.	"web"

- **Example:**


```
%SEARCH{"wiki" web="%USERSWEB%" scope="topic"}%
```

- **Example with format:**

```
%SEARCH{
  "FAQ"
  nonoise="on"
  header="| *Topic: * | *Summary: * |"
  format="| $topic      | $summary      |"
}%
```

(displays results in a table with header - [details](#))


Results are sorted alphanumerically on the web name (major key) and topic name (minor key). Only the minor key is affected by the `order` parameter.

 The appearance of the table emitted by the [SEARCH](#) may be controlled with [TablePlugin's](#) `%TABLE{}%` macro placed just before the `%SEARCH{}%`. Example: `%TABLE{ tablewidth="90%" }%`

**Related topics:** [FormattedSearch](#), [QuerySearch](#), [SearchHelp](#), [SearchPatternCookbook](#), [RegularExpression](#), [TOPICLIST](#), [WEBLIST](#)

## SERVERTIME{"format"} -- formatted server time

- Same format qualifiers as `%GMTIME%`
- Syntax: `%SERVERTIME%` OR `%SERVERTIME{"format"}%`
- `%SERVERTIME%` uses the Date format defined as `{DefaultDateFormat}` in [configure](#)
- Example: `%SERVERTIME{"$hou:$min"}%` expands to `21:53`

 **Note:** When used in a template topic, this macro will be expanded when the template is used to create a new topic. See [TemplateTopics#TemplateTopicsVars](#) for details.

- Related: [GMTIME](#), [SERVERTIME](#)

## SESSIONID -- unique ID for this session


- Syntax: `%SESSIONID%`
- Expand to: `e41c176e92ff6e3fe71db340f5ae71cb`
- Related: [UserAuthentication](#), [AUTHREALM](#), [SESSIONVAR](#), [LOGIN](#), [LOGOUT](#), [SESSION\\_VARIABLE](#)

## SESSIONVAR -- name of CGI and session variable that stores the session ID

- Syntax: `%SESSIONVAR%`
- Expand to: `FOSWIKISID`
- Related: [UserAuthentication](#), [AUTHREALM](#), [SESSIONID](#), [LOGIN](#), [LOGOUT](#), [SESSION\\_VARIABLE](#)

## SESSION\_VARIABLE -- get, set or clear a session variable

- Syntax: `%SESSION_VARIABLE{"name"}%`
- Syntax: `%SESSION_VARIABLE{"name" set="value"}%`
- Syntax: `%SESSION_VARIABLE{"name" clear=""}%`

 The users ID is in the `AUTHUSER` session variable, and is read-only

- Related: [UserAuthentication](#), [AUTHREALM](#), [SESSIONID](#), [SESSIONVAR](#), [LOGIN](#), [LOGOUT](#)

## SHOWPREFERENCE -- show where preferences are defined.

Preference values are shown in a bulleted list.

- `%SHOWPREFERENCE%`
  - Show all preferences
- `%SHOWPREFERENCE{"PREFERENCENAME"}%`
  - Show a single preference
  - Example:

```
%SHOWPREFERENCE{"ATTACHFILESIZELIMIT"}%
```

- Expands as:

```
* Set ATTACHFILESIZELIMIT = "10000"
* ATTACHFILESIZELIMIT was *finalised* in System.DefaultPreferences
```

- `%SHOWPREFERENCE{"PREFERENCENAME, PREFERENCENAME, ..."}%`
  - Show all of the preferences in a comma-separated list of preference names. This is particularly



useful for reviewing the access controls that apply to a topic.

- Example:

```
%SHOWPREFERENCE{"DENYWEBCHANGE,ALLOWWEBCHANGE"}%
```

- Expands as:

```
* Set DENYWEBCHANGE = ""
* Set ALLOWWEBCHANGE = "%USERSWEB%.AdminGroup"
* ALLOWWEBCHANGE was defined in System.WebPreferences
```


## SILVER -- start silver colored text

- `SILVER` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

**Example:**

```
%SILVER% silver text %ENDCOLOR%
```

**Expands to:** silver text

 `<color>` text must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

**Related:** [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)

## SKIN -- current skin

- `%SKIN%` expands the skin search path. For instance, `SKIN` can be set to `catskin`, `bearskin`.
- The `SKIN` setting can be overridden using the URL parameter `skin`, such as `?skin=catskin,bearskin`
- You can also extend the existing skin path using covers - see [COVER](#)
- Syntax: `%SKIN%`
- Expands to: `natedit,pattern`
- See [Skins](#) for more information
- Related: [SkinBrowser](#)

## SLIDESHOWEND -- end slideshow

- The `%SLIDESHOWEND%` macro is handled by the [SlideShowPlugin](#)
- Syntax: `%SLIDESHOWEND%`
- Example: See [SLIDESHOWSTART](#)
- Related: [SLIDESHOWSTART](#), [SlideShowPlugin](#)

## SLIDESHOWSTART -- convert a topic with headings into a slideshow

- The `%SLIDESHOWSTART%` macro is handled by the [SlideShowPlugin](#)
- Syntax: `%SLIDESHOWSTART%`
- Syntax: `%SLIDESHOWSTART{ template="MyOwnSlideTemplate" }%`
- Example:

```
%SLIDESHOWSTART%
```

```

---++ Sample Slide 1
  * Bullet 1
  * Bullet 2
---++ Sample Slide 2
  * Bullet 1
  * Bullet 2
%SLIDESHOWEND%

```

- Expands as:

## Start Presentation

### Slide 1: Sample Slide 1

- Bullet 1
- Bullet 2

### Slide 2: Sample Slide 2

- Bullet 1
- Bullet 2
- Related: [SLIDESHOWEND](#), [SlideShowPlugin](#)

## SPACEDTOPIC -- topic name, spaced and URL-encoded **deprecated**

- The current topic name with added URL-encoded spaces, for use in regular expressions that search for backlinks to the current topic
- Syntax: `%SPACEDTOPIC%`
- Expands to: `Var%20*SPACEDTOPIC`

**!** This is a deprecated macro. It can be duplicated with `%ENCODE{%SPACEOUT{"%TOPIC%" separator="*" }%}`

- Related: [SPACEOUT](#), [TOPIC](#), [ENCODE](#)

## SPACEOUT{"string"} -- renders string with spaces inserted in sensible places

- Inserts spaces after lower case letters that are followed by a digit or a capital letter, and after digits that are followed by a capital letter.
- Useful for spacing out [WikiWords](#)
- Syntax: `%SPACEOUT{ "%TOPIC%" }%`
- Expands to: `Macros`
- Supported parameters:


Parameter:	Description:	Default:
<code>separator</code>	The separator to put between words e.g. <code>%SPACEOUT{"DogsCatsBudgies"}</code> <code>separator=", " % -&gt; Dogs, Cats, Budgies</code>	<code>" "</code>

**💡** Spaced out WikiWords are not automatically linked. To `SPACEOUT` a WikiWord but preserve the link use "double bracket" format. For example, `[[WebHome]] [%SPACEOUT{"WebHome"}%]` expands to [Web Home](#)

- Related: [SPACEDTOPIC](#), [\\$PROPERSPACE\(\)](#) of [SpreadSheetPlugin](#)

## STARTINCLUDE -- start position of topic text if included

- If present in included topic, start to include text from this location up to the end, or up to the location of the `%STOPINCLUDE%` macro. A normal view of the topic shows everything except the `%STARTINCLUDE%` macro itself.
- Syntax: `%STARTINCLUDE%`


 If you want more than one part of the topic included, use `%STARTSECTION{type="include"}%` instead


- Related: [INCLUDE](#), [STARTSECTION](#), [STOPINCLUDE](#)

## STARTSECTION -- marks the start of a section within a topic

- Section boundaries are defined with `%STARTSECTION{ }%` and `%ENDSECTION{ }%`.
- Sections may be given a name to help identify them, and/or a type, which changes how they are used.
  - `type="section"` - the default, used for a generic section, such as a named section used by [INCLUDE](#).
  - `type="include"` - like `%STARTINCLUDE% ... %STOPINCLUDE%` except that you can have as many include blocks as you want which are all merged into one when included (`%STARTINCLUDE%` is restricted to only one). Sections of type include may not be given a name.
  - `type="expandvariables"` - all macros inside an "expandvariables" type section gets expanded when a new topic based on the template topic is created. See [TemplateTopics](#) for more information.
  - `type="templateonly"` - start position of text to be removed when a template topic is used. This is used to embed text that you *do not* want expanded when a new topic based on the template topic is created. See [TemplateTopics](#) for more information.
- Syntax: `%STARTSECTION{"name"}% ..... %ENDSECTION{"name"}%`
- Syntax: `%STARTSECTION{type="include"}% ..... %ENDSECTION{type="include"}%`
- Syntax: `%STARTSECTION{type="expandvariables"}% ... %ENDSECTION{type="expandvariables"}%`
- Syntax: `%STARTSECTION{type="templateonly"}% ..... %ENDSECTION{type="templateonly"}%`
- Supported parameters:

Parameter:	Description:	Default
"name"	Name of the section. Must be unique inside a topic.	Generated name
type="..."	Type of the section; type "section", "expandvariables", "include" or "templateonly"	"section"

 If a section is not given a name, it will be assigned one. Unnamed sections are assigned names starting with `_SECTION0` for the first unnamed section in the topic, `_SECTION1` for the second, etc..

 You can define nested sections. It is not recommended to overlap sections, although it is valid in Foswiki. Use named sections to make sure that the correct START and ENDS are matched. Section markers are **not** displayed when a topic is viewed.

- Related: [ENDSECTION](#), [INCLUDE](#), [NOP](#), [STARTINCLUDE](#), [STOPINCLUDE](#)

## STATISTICSTOPIC -- name of statistics topic

- Syntax: `%STATISTICSTOPIC%`
- Expands to: `Main.WebHome`, renders as [Main.WebHome](#)
- Related: [HOMETOPIC](#), [NOTIFYTOPIC](#), [TOPIC](#)

## STOPINCLUDE -- end position of topic text if included

- If present in included topic, stop to include text at this location and ignore the remaining text. A normal view of the topic shows everything except the `%STOPINCLUDE%` macro itself.

- Syntax: `%STOPINCLUDE%`
- Related: [INCLUDE](#), [STARTINCLUDE](#)

## SUBSCRIBE{ *attributes* } - subscribe the current user for notification of changes to the current topic

Anywhere in a topic or template:

- `%SUBSCRIBE{...}%` or simply `%SUBSCRIBE%`


Parameter	Default	Meaning
<code>who</code>	Logged-in user	Who to subscribe (wikiname, no web)
<code>topic</code>	Current topic	Topic to subscribe to. Wildcards may be used e.g. <code>topic="Item*"</code> will subscribe to all topics starting with <code>Item</code> . Use <code>topic="Main.*"</code> to subscribe to the WebNotify for that web.
<code>unsubscribe</code>	Not set	If set to "on", will <b>unsubscribe</b> the user
<code>format</code>	Simple HTML Link with an action anchor "Subscribe" or "Unsubscribe"	Defines custom look-and-feel for subscribe link using the following format variables: <code>\$topics</code> , <code>\$url</code> , <code>\$wikiname</code> , <code>\$action</code>
<code>formatunsubscribe</code>	<code>format</code>	Serves same function as <code>format</code> , but allows the user to define a different format than <code>subscribe</code> .

- The button toggles subscribe/unsubscribe if the user is already subscribed or vice versa. So after the next click on the button, the user will be unsubscribed.
- The plugin will do the best it can to optimise the subscription lists in [WebNotify](#).

## SYSTEMWEB -- name of documentation web

- The web containing all documentation and default [preference settings](#)
- Syntax: `%SYSTEMWEB%`
- Expands to: `System`
- Related: [USERSWEB](#)

## T -- tip icon

- Current value: T = 
- Related: [H](#), [I](#), [ICON](#), [M](#), [N](#), [P](#), [Q](#), [S](#), [U](#), [X](#), [Y](#)

## TAB{"text" ...} -- tab inside a tabpane widget

- Defines a tab inside a TABPANE area; must be closed using ENDTAB.
- Parameters:

Parameter:	Description:	Default:
<code>"text"</code>	label of the tab	Tab
<code>before</code>	when switching tabs, this is the javascript fragment to be executed just before the tab is displayed	
<code>after</code>	this javascript handler is to be executed after the tab has been made visible	
<code>afterload</code>	this javascript handler will be called when content loaded asynchronously (using the <code>url</code> parameter, below) has finished loading; depending on the network latency, this can be significantly later than execution of the <code>after</code> handler above	
<code>id</code>	id of this tab; this id can be used in the TABPANEs <code>select</code> parameter to	

	display this tab; this id is also added to the class attribute of the html element representing the tab button	
url	link from where to load the content of the tab asynchronously when selecting this tab; the result of the addressed handler will replace the content area; if no url is set the content of the TAB ... ENDTAB area will be shown when the tab is selected	
width	width of the tab area	auto
height	height of the tab area	auto
container	element where ajax content will be loaded; this is only used together with <code>url</code>	

- Related: [VarENDTAB](#), [VarTABPANE](#), [VarENDTABPANE](#), [jQueryPlugin](#), [jQueryTabpane](#)

## TABLE{ *attributes* } -- control attributes of tables and sorting of table columns

- The `%TABLE{}%` macro is handled by the [TablePlugin](#)
- Syntax: `%TABLE{ attributes }%`

### Attributes for tables

Argument	Comment	Default value	Example
<code>tableborder</code>	Table border width (pixels).	"1"	<code>tableborder="2"</code>
<code>tablebordercolor</code>	Table border color . Is only visible when <code>cellspacing</code> is larger than 1, or <code>cellborder</code> is 0, or <code>tablerules</code> is <code>none</code> , otherwise the cell borders overlap the table border.	unspecified	<code>tablebordercolor="#333"</code>
<code>tableframe</code>	Table frame, set to "void" (no sides), "above" (the top side only), "below" (the bottom side only), "hsides" (the top and bottom sides only), "lhs" (the left-hand side only), "rhs" (the right-hand side only), "vsides" (the right and left sides only), "box" (all four sides), "border" (all four sides).	unspecified	<code>tableframe="hsides"</code>
<code>tablerules</code>	Table rules, set to "none" (no rules), "groups" (rules will appear between row groups and column groups only), "rows" (rules will appear between rows only), "cols" (rules will appear between columns only), "all" (rules will appear between all rows and columns). See also: <code>headerrules</code> and <code>datarules</code> .	unspecified	<code>tablerules="rows"</code>
<code>tablewidth</code>	Table width: percentage of window width, or absolute pixel value.	unspecified	<code>tablewidth="100%"</code>
<code>headerrows</code>	Number of header rows to exclude from sort. (will be rendered in a HTML <code>thead</code> section)	"1"	<code>headerrows="1"</code>
<code>footerrows</code>	Number of footer rows to exclude from sort. (will be rendered in a HTML <code>tfoot</code> section)	"0"	<code>footerrows="1"</code>
<code>id</code>	Unique table identifier string, used for	<code>tableN</code> (where N	<code>id="userTable"</code>

	targeting a table with CSS.	is the table order number on the page)	
summary	Table summary used by screen readers: A summary of what the table presents. It should provide an orientation for someone who listens to the table.	unspecified	summary="List of subscribed users"
caption	Table caption: A title that will be displayed just above the table.	unspecified	caption="Users"
inlinemarkup	Set to "on" to generate inline markup HTML (in addition to the CSS markup); useful if you need to copy the table, for instance to paste the table into an email).	unspecified	inlinemarkup="on"

### Attributes for table sorting

Argument	Comment	Default value	Example
sort	Set the table sorting user interface (clickable column headers) "on" or "off".	unspecified	sort="on"
initsort	Column to sort initially (use "1" for the first column). If specified, sorting is enabled; by setting <code>sort="off"</code> the sorting interface can be hidden.	unspecified	initsort="2"
initdirection	Initial sorting direction for <code>initsort</code> , set to "up" (descending, or decreasing in value) or "down" (ascending, or increasing in value).	down	initdirection="up"
disableallsort	Disable all sorting, both <code>initsort</code> and header sort. This is mainly used by plugins such as the <a href="#">EditTablePlugin</a> to disable sorting in a table while editing the table.	unspecified	disableallsort="on"

### Attributes for table cells

Argument	Comment	Default value	Example
cellpadding	Cell padding (pixels).	unspecified	cellpadding="0"
cellspacing	Cell spacing (pixels).	unspecified	cellspacing="3"
cellborder	Cell border width (pixels).	unspecified	cellborder="0"
valign	Vertical alignment of cells and headers, set to "top", "middle", "bottom" or "baseline".	unspecified	valign="top"
columnwidths	Column widths: Comma delimited list of column widths, percentage or absolute pixel value.	unspecified	columnwidths="80%,20%"

### Attributes for data cells

Argument	Comment	Default value	Example
----------	---------	---------------	---------

<code>datarules</code>	Set to <code>"none"</code> (no rules), <code>"rows"</code> (rules will appear between rows only), <code>"cols"</code> (rules will appear between columns only), <code>"all"</code> (rules will appear between all rows and columns). Overrides <code>tablerules</code> for data cells.	unspecified	<code>datarules="none"</code>
<code>datavalign</code>	Vertical alignment of data cells; overrides <code>valign</code> .	unspecified	<code>datavalign="top"</code>
<code>dataalign</code>	Data cell alignment, one value for all columns, or a comma separated list for different alignment of individual columns. Set to <code>"left"</code> , <code>"center"</code> , <code>"right"</code> or <code>"justify"</code> . Overrides individual cell settings.	unspecified	<code>dataalign="center"</code>
<code>databg</code>	Data cell background colour, a comma separated list. Specify <code>"none"</code> for no colour, that is to use the colour/background of the page the table is on.	<code>"#edf4f9, #fff"</code>	<code>databg="#f2f2f2, #fff"</code>
<code>databgsorted</code>	Data cell background colour of a sorted column; see <code>databg</code> .	the values of <code>databg</code>	<code>databgsorted="#d4e8e4, #e5f5ea"</code>
<code>datacolor</code>	Data cell text colour, a comma separated list.	unspecified	<code>datacolor="#00c, #000"</code>

## Attributes for headers

Argument	Comment	Default value	Example
<code>headerrules</code>	Set to <code>"none"</code> (no rules), <code>"rows"</code> (rules will appear between rows only), <code>"cols"</code> (rules will appear between columns only), <code>"all"</code> (rules will appear between all rows and columns). Overrides <code>tablerules</code> for header cells.	unspecified	<code>headerrules="none"</code>
<code>headerbg</code>	Header cell background colour. Specify <code>"none"</code> for no colour, that is to use the colour/background of the page the table is on.	<code>"#6b7f93"</code>	<code>headerbg="#999"</code>
<code>headerbgsorted</code>	Header cell background colour of a sorted column. Specify <code>"none"</code> for no colour, that is to use the colour/background of the page the table is on.	the value of <code>headerbg</code>	<code>headerbgsorted="#32596c"</code>
<code>headercolor</code>	Header cell text colour.	<code>"#fff"</code>	<code>headercolor="#00c"</code>
<code>headervalign</code>	Vertical alignment of header cells; overrides <code>valign</code> .	unspecified	<code>headervalign="top"</code>
<code>headeralign</code>	Header cell alignment, one value for all columns, or a comma separated list for different alignment of individual columns. Set to <code>"left"</code> , <code>"center"</code> , <code>"right"</code> or <code>"justify"</code> . Overrides individual cell settings.	unspecified	<code>headeralign="left, right"</code>

## Other attributes

Argument	Comment	Default value	Example
include	Other topic defining the <code>TABLE</code> parameters. The first <code>%TABLE%</code> in the topic is used. This is useful if you have many topics with the same table format and you want to update the format in one place. Use <code>topic</code> or <code>web.topic</code> notation.	unspecified	<code>include="Main.WebHome"</code>

- Example:

```
%TABLE{ tableborder="0" cellpadding="4" cellspacing="3" cellborder="0" }%
| *A1* | *B1* |
| A2   | B2   |
```

- Expands as:

```
A1 B1
```

```
A2 B2
```

- Related: See [TablePlugin](#) for more details

## TABPANE{...} -- tabpane widget

- This macro starts the tabpane, containing a series of `TAB...ENDTABs` and ends with `ENDTABPANE`. A complete tabpane normally looks like this:

```
%TABPANE%
%TAB{"tab 1"}%
...
%ENDTAB%
%TAB{"tab 2"}%
...
%ENDTAB%
%ENDTABPANE%
```

Multiple tabpanes can be nested using the following scheme:

```
%TABPANE%
%TAB{"tab 1"}%
%TABPANE%
%TAB{"tab 1.1"}%
...
%ENDTAB%
%TAB{"tab1.2"}%
...
%ENDTAB%
%ENDTABPANE%
%ENDTAB%
%TAB{"tab 2"}%
...
%ENDTAB%
%ENDTABPANE%
```



- Example: see [jQueryTabpane](#) for more examples
- Parameters:

Parameter:	Description:	Default:
select	number or id of tab to select	1
automaxexpand	resizes the tabpane to the maximum height to fit into the window	off
minheight	when automaxexpand is enabled, this is the minimum size a tab is allowed to be resized	230
class	extra class: use <code>simple</code> for a non-3D tabpane	
animate	(on/off) enables/disables animation of switching tabs	off

- Related: [VarTAB](#), [VarENDTAB](#), [VarENDTABPANE](#), [jQueryPlugin](#), [jQueryTabpane](#)


## TEAL -- start teal colored text

- `TEAL` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

### Example:

```
%TEAL% teal text %ENDCOLOR%
```

### Expands to: teal text

 `%<color>% text` must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

Related: [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)

## TOC{"Topic"} -- table of contents


- Table of Contents. Shows a TOC that is generated automatically based on headings of a topic. Headings in [WikiSyntax](#) ("`----+ text`") and HTML ("`<h2>text</h2>`") are taken into account. Any heading text after "`!!`" is excluded from the TOC; for example, write "`----+!! text`" if you do not want to list a header in the TOC
- Syntax: `%TOC%` OR `%TOC{"SomeTopic" ...}%`
- Supported parameters:

Parameter:	Description:	Default:
"TopicName"	topic name	Current topic
web="Name"	Name of web	Current web
depth="2"	Limit depth of headings shown in TOC	6
title="Some text"	Title to appear at top of TOC	none
id="Some_id"	Optional ID in case multiple TOCs are on the page and each TOC needs to be addressable with an anchor link. Allowed characters: a-zA-Z0-9-_, no spaces.  If you don't specify an id, the anchor <code>foswikiTOC</code> can be used in a link: <code>[[#foswikiTOC][Back to TOC]]</code> creates <a href="#">Back to TOC</a> .	"foswikiTOC"

- Default settings are defined in [DefaultPreferences](#), and can be overridden in [SitePreferences](#)
- Examples:

```
%TOC{depth="2"}%
%TOC{"CompleteDocumentation" web="%SYSTEMWEB%" title="Contents:"}%
```

See also: [Foswiki:Support/HowToCreateATableOfContents](#)

 If multiple headers have the exact same text, the anchors for the 2nd, 3rd etc will be suffixed by `_AN1`, `_AN2` etc so the anchors become unique.

## TOPIC -- name of current topic

- `%TOPIC%` expands to the name of the topic. If you are looking at the text of an included topic, it is the name of the included topic.
- Syntax: `%TOPIC%`
- Expands to: `Macros`, renders as `Macros`
- Related: [BASETOPIC](#), [INCLUDINGTOPIC](#), [TOPICLIST](#), [WEB](#)

## TOPICLIST{"format"} -- topic index of a web

- List of all topics in a web. The "format" defines the format of one topic item. It may include formatting tokens: The `$topic` token gets expanded to the topic name, `$marker` to `marker` parameter where topic matches `selection`, and `$web` to the name of the web, or any of the standard [FormatTokens](#).
- Syntax: `%TOPICLIST{"format" ...}%`
- Supported parameters:

Parameter:	Description:	Default:
"format"	Format of one line, may include <code>\$web</code> (name of web), <code>\$topic</code> (name of the topic), <code>\$marker</code> (which expands to <code>marker</code> for the item matching <code>selection</code> only)	"\$topic"
format="format"	(Alternative to above)	"\$topic"
separator=", "	line separator	"\$n" (new line)
marker="selected"	Text for <code>\$marker</code> if the item matches <code>selection</code>	"selected"
selection="TopicA, TopicB"	Current value to be selected in list	(none)
web="Name"	Name of web	Current web

- Examples:

```
Create a bullet list of all topics:
%TOPICLIST{" * $web.$topic"}%

Create a comma separated list of all topics:
%TOPICLIST{separator=", "}%

Create an option list (for drop down menus):
%TOPICLIST{" <option>$topic</option>"}%


Create an option list of web topics with the current topic selected:
<select>%TOPICLIST{
  " <option $marker value='$topic'>$topic</option>"
  separator=" "
  selection="%TOPIC%"
}%</select>
```

- Related: [SEARCH](#), [WEBLIST](#)

## TOPICURL -- shortcut to viewing the current topic

- This macro always expands to an url pointing to the current base topic.
- Current value: `TOPICURL` = <http://foswiki.trunk/bin/view/System/Macros>

## TWIKIWEB -- synonym for SYSTEMWEB

 **Deprecated.** Use `%SYSTEMWEB%` instead

## TWISTY

This renders the button as well as the toggled content section contained within this and the closing `ENDTWISTY` tag.

Usage: `%TWISTY{ ... }% Toggable contents %ENDTWISTY%`

Parameter	Value	Description	Remark
<code>id</code>	Unique identifier	Used to link <a href="#">TWISTYBUTTON</a> and <a href="#">TWISTYTOGGLE</a>	optional
<code>link</code>	Link label	Link label for both show and hide links	optional
<code>hidelink</code>	Link label	Hide link label	optional
<code>showlink</code>	Link label	Show link label	optional
<code>mode</code>	"div" or "span"	Specify if the Twisty Toggle section will use a <code>&lt;div&gt;</code> or a <code>&lt;span&gt;</code> tag. Note that if the contents contains block elements such as <code>div</code> , <code>mode</code> should be <code>div</code> as well to create valid HTML markup.	optional, defaults to <code>&lt;div&gt;</code>
<code>showimgleft</code>	Image url	Specify the url of an image that will be displayed with the show link at the left side of the link. You may use <a href="#">ICONURLPATH</a> to display one of the <a href="#">DocumentGraphics</a> icons. Alternatively use an image attached to the topic.	optional, defaults to no image
<code>hideimgleft</code>	Image url	Specify the url of an image that will be displayed with the hide link at the left side of the link. You may use <a href="#">ICONURLPATH</a> to display one of the <a href="#">DocumentGraphics</a> icons. Alternatively use an image attached to the topic.	optional, defaults to no image
<code>showimgright</code>	Image url	Specify the url of an image that will be displayed with the show link at the right side of the link. You may use <a href="#">ICONURLPATH</a> to display one of the <a href="#">DocumentGraphics</a> icons. Alternatively use an image attached to the topic.	optional, defaults to no image
<code>hideimgright</code>	Image url	Specify the url of an image that will be displayed with the hide link at the right side of the link. You may use <a href="#">ICONURLPATH</a> to display one of the <a href="#">DocumentGraphics</a> icons. Alternatively use an image attached to the topic.	optional, defaults to no image
<code>remember</code>	"on", "off"	If "on", the Twisty state is remembered the next time the page is shown. If "off", the stored setting will be cleared.	optional, no default

 **Note:** when used, think carefully about a unique name (`id`) for the Twisty, otherwise the cookie that is set might affect other

Twisties with the same name. Also note that only interaction is stored, not the state of the Twisty when left unclicked.

<code>start</code>	"hide" or "show"	Initial state of the Twisty; this will override any setting stored in a cookie (see <code>remember</code> ).	optional, default no initial state
<code>firststart</code>	"hide" or "show"	Initial state of the Twisty the first time the visitor gets to see the Twisty; this will NOT override cookie settings (see <code>remember</code> ).	optional, default no initial state
<code>noscript</code>	"hide"	Make content hidden in case use does not have JavaScript on	optional, default content is shown in case JavaScript if off
<code>class</code>	CSS class name	Class for Twisty div or span	optional, default none
<code>linkclass</code>	CSS class name	Class for link	optional, default none
<code>prefix</code>	Text	Text to display before the show/hide links	optional, default none
<code>suffix</code>	Text	Text to display after the show/hide links	optional, default none
<code>img</code>	Image url	Deprecated, use <code>showimgleft</code> , <code>hideimgleft</code> , <code>showimgright</code> or <code>hideimgright</code> .	optional, defaults to no image
<code>imgleft</code>	Image url	Deprecated, use <code>showimgleft</code> , <code>hideimgleft</code> , <code>showimgright</code> or <code>hideimgright</code> .	optional, defaults to no image
<code>imgright</code>	Image url	Deprecated, use <code>showimgleft</code> , <code>hideimgleft</code> , <code>showimgright</code> or <code>hideimgright</code> .	optional, defaults to no image
<code>hideimg</code>	Image url	Deprecated, use <code>showimgleft</code> , <code>hideimgleft</code> , <code>showimgright</code> or <code>hideimgright</code> .	optional, defaults to no image
<code>showimg</code>	Image url	Deprecated, use <code>showimgleft</code> , <code>hideimgleft</code> , <code>showimgright</code> or <code>hideimgright</code> .	optional, defaults to no image

- Related: [VarENDTWISTY](#), [VarENDTWISTYTOGGLE](#), [VarTWISTY](#), [VarTWISTYBUTTON](#), [VarTWISTYHIDE](#), [VarTWISTYSHOW](#), [VarTWISTYTOGGLE](#)

## TWISTYBUTTON

**Shorthand version for TWISTYSHOW & TWISTYHIDE** This is useful if both the show and the hide button take the same arguments.

- Supported parameters: all parameters supported by [TWISTY](#), except for `noscript` and `class` (only used for 'toggle' content)
- Parameter differences:
  - `mode`: button mode defaults to `div`
- Syntax: `%TWISTYBUTTON{id="myid" ... }%`
- Supported parameters:

Parameter	Value	Description	Remark
mode	"div" or "span"	Specify if the <b>Twisty button</b> will use a <code>&lt;div&gt;</code> or a <code>&lt;span&gt;</code> tag. Note that if the contents contains block elements such as <code>div</code> , <code>mode</code> should be <code>div</code> as well to create valid HTML markup.	optional, defaults to <code>&lt;div&gt;</code>

- Example:

```
%TWISTYBUTTON{
  id="myid"
  link="more"
}%%TWISTYTOGGLE{
  id="myid"
}%content%ENDTWISTYTOGGLE%
```

- Expands as:  
content
- Related: [VarENDTWISTY](#), [VarENDTWISTYTOGGLE](#), [VarTWISTY](#), [VarTWISTYBUTTON](#), [VarTWISTYHIDE](#), [VarTWISTYSHOW](#), [VarTWISTYTOGGLE](#)

## TWISTYHIDE

### Hide/close link

- Syntax: `%TWISTYHIDE{id="myid" ... }%`
- Supported parameters:

Parameter	Value	Description	Remark
id	Unique identifier	Used to link <a href="#">TWISTYSHOW</a> , <a href="#">TWISTYHIDE</a> and <a href="#">TWISTYTOGGLE</a>	required
link	Link label	Hide link label	optional
mode	"div" or "span"	Specify if the Twisty Hide link will use a <code>&lt;div&gt;</code> or a <code>&lt;span&gt;</code> tag. Note that if the contents contains block elements such as <code>div</code> , <code>mode</code> should be <code>div</code> as well to create valid HTML markup.	optional, defaults to <code>&lt;div&gt;</code>
img	Image url	Specify the url of an image that will be displayed at the <b>right side</b> of the link. You may use <a href="#">ICONURLPATH</a> to display one of the <a href="#">DocumentGraphics</a> icons. Alternatively use an image attached to the topic.	optional, defaults to no image
remember	"on", "off"	If "on", the Twisty state is remembered the next time the page is shown. If "off", the stored setting will be cleared. <b>Note:</b> when used, think carefully about a unique name (id) for the Twisty, otherwise the cookie that is set might affect other Twisties with the same name. Also note that only interaction is stored, not the state of the Twisty when left unclicked.	optional, no default
start	"hide" or "show"	Initial state of the Twisty; this will override any setting stored in a cookie (see <code>remember</code> ).	optional, default no initial state
firststart	"hide" or "show"	Initial state of the Twisty the first time the visitor gets to see the Twisty; this will NOT override cookie settings (see <code>remember</code> ).	optional, default no initial state

- Example:

```
%TWISTYHIDE{id="demo" link=" Click to Fold " imgleft="%ICONURLPATH{toggleclose}%"}
```

- Related: [VarENDTWISTY](#), [VarENDTWISTYTOGGLE](#), [VarTWISTY](#), [VarTWISTYBUTTON](#), [VarTWISTYHIDE](#), [VarTWISTYSHOW](#), [VarTWISTYTOGGLE](#)

## TWISTYSHOW

### Show/open link

- **Syntax:** `%TWISTYSHOW{id="myid" ... }%`
- Supported parameters:

Parameter	Value	Description	Remark
<code>id</code>	Unique identifier	Used to link <a href="#">TWISTYSHOW</a> , <a href="#">TWISTYHIDE</a> and <a href="#">TWISTYTOGGLE</a>	required
<code>link</code>	Link label	Show link label	optional
<code>mode</code>	"div" or "span"	Specify if the Twisty Show link will use a <code>&lt;div&gt;</code> or a <code>&lt;span&gt;</code> tag. Note that if the contents contains block elements such as <code>div</code> , <code>mode</code> should be <code>div</code> as well to create valid HTML markup.	optional, defaults to <code>&lt;div&gt;</code>
<code>img</code>	Image url	Specify the url of an image that will be displayed at the <b>right side</b> of the link. You may use <a href="#">ICONURLPATH</a> to display one of the <a href="#">DocumentGraphics</a> icons. Alternatively use an image attached to the topic.	optional, defaults to no image
<code>imgleft</code>	Image url	Specify the url of an image that will be displayed at the left side of the link. You may use <a href="#">ICONURLPATH</a> to display one of the <a href="#">DocumentGraphics</a> icons. Alternatively use an image attached to the topic.	optional, defaults to no image
<code>imgright</code>	Image url	Specify the url of an image that will be displayed at the right side of the link. You may use <a href="#">ICONURLPATH</a> to display one of the <a href="#">DocumentGraphics</a> icons. Alternatively use an image attached to the topic.	optional, defaults to no image
<code>remember</code>	"on", "off"	If "on", the Twisty state is remembered the next time the page is shown. If "off", the stored setting will be cleared. <b>Note:</b> when used, think carefully about a unique name (id) for the Twisty, otherwise the cookie that is set might affect other Twisties with the same name. Also note that only interaction is stored, not the state of the Twisty when left unclicked.	optional, no default
<code>start</code>	"hide" or "show"	Initial state of the Twisty; this will override any setting stored in a cookie (see <code>remember</code> ).	optional, default no initial state
<code>firststart</code>	"hide" or "show"	Initial state of the Twisty the first time the visitor gets to see the Twisty; this will NOT override cookie settings (see <code>remember</code> ).	optional, default no initial state

- Example:

```
%TWISTYSHOW{id="demo" link=" Click to Unfold " imgleft="%ICONURLPATH{toggleopen}%"}
```

- Related: [VarENDTWISTY](#), [VarENDTWISTYTOGGLE](#), [VarTWISTY](#), [VarTWISTYBUTTON](#), [VarTWISTYHIDE](#), [VarTWISTYSHOW](#), [VarTWISTYTOGGLE](#)

# TWISTYTOGGLE

## Twisty Toggle contents section

- Syntax: `%TWISTYTOGGLE{id="myid"}%`
- Supported parameters:

Parameter	Value	Description	Remark
<code>id</code>	Unique identifier	Used to link <a href="#">TWISTYSHOW</a> , <a href="#">TWISTYHIDE</a> and <a href="#">TWISTYTOGGLE</a> .	required
<code>mode</code>	"div" or "span"	Specify if the Twisty Toggle section will use a <code>&lt;div&gt;</code> or a <code>&lt;span&gt;</code> tag. Note that if the contents contains block elements such as <code>div</code> , <code>mode</code> should be <code>div</code> as well to create valid HTML markup.	optional, defaults to <code>&lt;div&gt;</code>
<code>class</code>	CSS class name	Class for content div or span	optional, default none
<code>linkclass</code>	CSS class name	Class for link	optional, default none
<code>remember</code>	"on", "off"	If "on", the Twisty state is remembered the next time the page is shown. If "off", the stored setting will be cleared. <b>Note:</b> when used, think carefully about a unique name (id) for the Twisty, otherwise the cookie that is set might affect other Twisties with the same name. Also note that only interaction is stored, not the state of the Twisty when left unclicked.	optional, no default
<code>start</code>	"hide" or "show"	Initial state of the Twisty; this will override any setting stored in a cookie (see <code>remember</code> ).	optional, default no initial state
<code>firststart</code>	"hide" or "show"	Initial state of the Twisty the first time the visitor gets to see the Twisty; this will NOT override cookie settings (see <code>remember</code> ).	optional, default no initial state
<code>noscript</code>	"hide"	Make content hidden in case use does not have JavaScript on	optional, default content is shown with no JavaScript

- Example:

```
%TWISTYTOGGLE{id="demo" mode="div" remember="on"}%My content%ENDTWISTYTOGGLE%
```

- Related: [VarENDTWISTY](#), [VarENDTWISTYTOGGLE](#), [VarTWISTY](#), [VarTWISTYBUTTON](#), [VarTWISTYHIDE](#), [VarTWISTYSHOW](#), [VarTWISTYTOGGLE](#)

## U -- "updated" icon

- Current value: U = **UPDATED**
- Related: [H](#), [I](#), [ICON](#), [M](#), [N](#), [P](#), [Q](#), [S](#), [T](#), [X](#), [Y](#)

## URLPARAM{"name"} -- get value of a URL parameter

- Returns the value of a URL parameter.
- Syntax: `%URLPARAM{"name"}%`
- Supported parameters:

Parameter:	Description:	Default:
"name"	The name of a URL parameter	required
default="..."	Default value in case parameter is empty or missing	empty string
newline=" "	Convert newlines in textarea to other delimiters	no conversion
encode="off" encode="entity" encode="safe" encode="url" encode="quote"	Control how special characters are encoded <hr/> off: No encoding. Avoid using this when possible. See the security warning below. <hr/> entity: Encode special characters into HTML entities. See <a href="#">ENCODE</a> for more details. <hr/> safe: Encode characters '"<>% into HTML entities. <hr/> url: Encode special characters for URL parameter use, like a double quote into %22 <hr/> quote: Escape double quotes with backslashes (\"), does not change other characters; required when feeding URL parameters into other macros.	"safe"
multiple="on" multiple="[[ \$item ]]"	If set, gets all selected elements of a <select multiple="multiple"> tag. A format can be specified, with \$item indicating the element, e.g. multiple="Option: \$item" (also supports the standard <a href="#">format tokens</a> )	first element
separator=", "	Separator between multiple selections. Only relevant if multiple is specified	"\n" (new line)

- **Example:** `%URLPARAM{"skin"}%` returns `print` for a `.../view/System/Macros?skin=print` URL
- **Notes:**
  - URL parameters passed into HTML form fields must be entity [ENCODEd](#).
  - Double quotes in URL parameters must be escaped when passed into other macros.  
Example: `%SEARCH{ "%URLPARAM{ "search" encode="quote" }%" noheader="on" }%`
  - When used in a template topic, this macro will be expanded when the template is used to create a new topic. See [Template Topics#Template Topics Vars](#) for details.
  - Watch out for internal parameters, such as `rev`, `skin`, `template`, `topic`, `web`; they have a special meaning in Foswiki. Common parameters and view script specific parameters are documented at [CommandAndCGIScripts](#).
  - If you have `%URLPARAM{` in the value of a URL parameter, it will be modified to `%<nop>URLPARAM{`. This is to prevent an infinite loop during expansion.
  - Security warning! Using URLPARAM can easily be misused for cross-site scripting unless specific characters are entity encoded. By default URLPARAM encodes the characters `'"<>%` into HTML entities (same as `encode="safe"`) which is relatively safe. The safest is to use `encode="entity"`. When passing URLPARAM inside another macro always use double quotes ("" ) combined with using URLPARAM with `encode="quote"`. For maximum security against cross-site scripting you are advised to install the [Foswiki:Extensions.SafeWikiPlugin](#).
- Related: [ENCODE](#), [SEARCH](#), [FormattedSearch](#), [QUERYSTRING](#)

## USERINFO{"name" format=""} -- retrieve details about a user

- **Syntax:** `%USERINFO%`
- **Expands to:** `guest`, `Main.WikiGuest`, (lists `$username`, `$wikiusername`, `$emails`)
- **Supported parameters:**
  - `"name"` - Wikiname or login name. May be a group. Defaults to the current user
  - `format="..."` - Format string; see below for supported formatting tokens. Defaults to `$username`,



`$wikiusername`, `$emails`

- `$emails` - Comma separated list of email addresses known to the user mapper (this would normally be [TopicUserMappingContrib](#)). If expanding for a group, then this will be the email addresses of all members.
  - `$username` - Login name. If expanding for a group, this should expand as `unknown`.
  - `$wikiname`, `$wikiusername` - Wikiname and `Main.Wikiname`, respectively.
  - `$groups` - Comma separated list of group membership. Currently only expands for users.
  - `$isadmin` - Has admin privileges (expands to `true` or `false`).
  - `$isgroup` - Is a group (expands to `true` or `false`).
  - The [standard format tokens](#) are also supported.
- Examples
    - With formatted output, using tokens:


```
%USERINFO{ format="$username is really $wikiname" }%
```

Expands to: `guest is really WikiGuest`


- Retrieve information about another user:

```
%USERINFO{ "WikiGuest" format="$username is really $wikiname" }%
```

Expands to: `guest is really WikiGuest`

 You need to be a member of [AdminGroup](#) for the `USERINFO` macro to display details about other users


- Notes

 Use wikinames when specifying a `name`. You can also pass a login name. You can only see information about another user if you are an admin, or the `{AntiSpam}{HideUserDetails}` configuration option is not enabled. (*User details are hidden on this site*)

- Related: [USERNAME](#), [WIKINAME](#), [WIKIUSERNAME](#), [GROUPINFO](#), [UserAuthentication](#), [ChangeEmailAddress](#)

## USERNAME -- your login username

- Foswiki makes names available in three formats: `USERNAME` like `jsmith`, `WIKINAME` like `JohnSmith` and `WIKIUSERNAME` like `Main.JohnSmith`. Un-authenticated users are all [WikiGuest](#).
- Syntax: `%USERNAME%`
- Expands to: `guest`

 When used in a template topic, this macro will be expanded when the template is used to create a new topic. See [TemplateTopics#TemplateTopicsVars](#) for details

- Related: [REMOTE\\_USER](#), [USERINFO](#), [WIKINAME](#), [WIKIUSERNAME](#), [UserAuthentication](#)

## USERSWEB -- name of users web

- The web containing individual user topics, [WikiGroups](#), and customised site-wide preferences.
- Syntax: `%USERSWEB%`
- Expands to: `Main`
- Related: [SYSTEMWEB](#)

## VAR{"NAME" web="Web"} -- get a preference value from another web

- Syntax: `%VAR{"NAME" web="Web"}%`
- Example: To get `%WEBBGOLOR%` of the Main web write `%VAR{"WEBBGOLOR" web="Main"}%`, which expands to `#FFFA6`
- Related: [WEBPREFSTOPIC](#)

## VBAR -- vertical bar

- The vertical bar macro can be used in TML tables.
- Current value: `VBAR = |`
- Related: [BR](#), [BULLET](#), [BB](#), [BB2](#), [BB3](#), [BB4](#), [CARET](#)

## WEB -- name of current web

- `%WEB%` expands to the name of the web where the topic is located. If you are looking at the text of an included topic, it is the web where the included topic is located.
- Syntax: `%WEB%`
- Expands to: `System`
- Related: [BASEWEB](#), [INCLUDINGWEB](#), [TOPIC](#)

## WEBLIST{"format"} -- index of all webs

- List of all webs. Obfuscated webs are excluded, e.g. webs with a `NOSEARCHALL = on` [preference setting](#). The `"format"` defines the format of one web item. The `$name` gets expanded to the name of the web, `$qname` gets expanded to double quoted name, `$marker` to `marker` where web matches `selection`. Subwebs are listed recursively.
- Syntax: `%WEBLIST{"format" ...}%`
- Supported parameters:

Parameter:	Description:	Default:
<code>"format"</code>	Format of one line, may include <code>\$name</code> (the name of the web), <code>\$qname</code> (the name of the web in double quotes), <code>\$indentedname</code> (the name of the web with parent web names replaced by indents, for use in indented lists), and <code>\$marker</code> (which expands to <code>marker</code> for the item matching <code>selection</code> only)	<code>\$name</code>
<code>format="format"</code>	(Alternative to above)	<code>\$name</code>
<code>separator=", "</code>	Line separator	<code>\$n (new line)</code>
<code>web=""</code>	if you specify <code>\$web</code> in format, it will be replaced with this value.	none
<code>webs="public"</code>	Comma separated list of webs to consider. This list can include two pseudo-webs, <code>public</code> which expands to all non-hidden and <code>webtemplate</code> which expands to the names of all template webs. <b>NOTE:</b> Administrators will see all webs, not just the public ones	<code>public</code>
<code>subwebs="Sandbox"</code>	Specifies a single web. If specified, then <code>public</code> and <code>webtemplate</code> (described above) will expand relative to show subwebs *below this web only.	<code>""</code>
<code>selection="%WEB%"</code>	Entry to be selected in list. If one of the webs matches this selection, then <code>\$marker</code> in the <code>format</code> will be expanded	<code>selection=%WEB%</code>
<code>marker="selected"</code>	Text for <code>\$marker</code> if the item matches <code>selection</code>	<code>selected="selected"</code>

- Example, creates a bullet list of all webs:

```
%WEBLIST{" * [[${name.%HOMETOPIC%}]]"}%
```

- Example, creates a dropdown of all public webs + Trash web, with the current web highlighted:

```
<form><select name="web">%WEBLIST{
  "<option $marker value='${qname}'>$name</option>"
  webs="Trash, public"
  selection="%WEB%"
  separator=" "
}% </select></form>
```

**!** WEBLIST will not show a web called 'TWiki' even if it exists in the file system unless the TWikiCompatibilityPlugin is installed and activated in [configure](#). This is done to ensure that the TWiki compatibility components such as the TWiki web are only visible and active when needed

- Related: [TOPICLIST](#), [SEARCH](#)

## WEBPREFSTOPIC -- name of web preferences topic

- Syntax: %WEBPREFSTOPIC%
- Expands to: [WebPreferences](#), renders as [WebPreferences](#)
- Related: [HOMETOPIC](#), [WIKIPREFSTOPIC](#), [VAR](#)

## WHITE -- start white colored text

- WHITE is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

### Example:

```
%WHITE% white text %ENDCOLOR%
```

### Expands to:

**!** %<color>% text must end with %ENDCOLOR%. If you want to switch from one color to another one you first need to end the active color with %ENDCOLOR%, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

Related: [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)

## WIKIHOMEURL -- site home URL


- Syntax %WIKIHOMEURL%
- Expands to `/bin/view`
- Normally by default set to %SCRIPTURLPATH("view")%

**!** For the top bar logo URL use %WIKILOGOURL% defined in [WebPreferences](#) instead.

- Related: [WIKITOOLNAME](#)

## WIKINAME -- your Wiki username

- The [WikiName](#) is the same as `%USERNAME%` if not defined in the [WikiUsers](#) topic
- Syntax: `%WIKINAME%`
- Expands to: `WikiGuest`

 When used in a template topic, this macro will be expanded when the template is used to create new topic. See [TemplateTopics#TemplateTopicsVars](#) for details

- Related: [REMOTE\\_USER](#), [USERINFO](#), [USERNAME](#), [WIKIUSERNAME](#), [WIKIUSERSTOPIC](#)

## WIKIPREFSTOPIC -- name of site-wide preferences topic


- Syntax: `%WIKIPREFSTOPIC%`
- Expands to: `DefaultPreferences`, renders as [DefaultPreferences](#)
- Related: [HOMETOPIC](#), [WEBPREFSTOPIC](#)

## WIKITOOLNAME -- name of your site

- Syntax: `%WIKITOOLNAME%`
- Expands to: `Foswiki`
- Related: [WIKIHOMEURL](#), [WIKIVERSION](#)

## WIKIUSERNAME -- your Wiki username with web prefix

- Your `%WIKINAME%` with Main web prefix, useful to point to your Foswiki home page
- Syntax: `%WIKIUSERNAME%`
- Expands to: `Main.WikiGuest`, renders as [WikiGuest](#)

 When used in a template topic, this macro will be expanded when the template is used to create a new topic. See [TemplateTopics#TemplateTopicsVars](#) for details

- Related: [REMOTE\\_USER](#), [USERINFO](#), [USERNAME](#), [WIKINAME](#)

## WIKIUSERSTOPIC -- name of topic listing all registered users

- Syntax: `%WIKIUSERSTOPIC%`
- Expands to: `WikiUsers`, with Main prefix renders as [WikiUsers](#)
- Related: [WIKIUSERNAME](#)

## WIKIVERSION -- the version of the installed Foswiki engine

- Syntax: `%WIKIVERSION%`
- Expands to: `Foswiki-1.2.0-alpha, Sat, 19 May 2012, build 14862`
- Related: [PLUGINVERSION](#), [WIKITOOLNAME](#)

## WIKIWEBMASTER -- feedback email address for site


- Syntax: `%WIKIWEBMASTER%`
- Expands to: `arthur@visiblearea.com`
- Related: [WIKIWEBMASTERNAME](#)

## WIKIWEBMASTERNAME -- Name of the administrator for the site


- Syntax: `%WIKIWEBMASTERNAME%`

- Expands to: Arthur Clemens
- Related: [WIKIWEBMASTER](#)

## X -- warning icon

- Current value: X = 
- Related: [H](#), [I](#), [ICON](#), [M](#), [N](#), [P](#), [Q](#), [S](#), [T](#), [U](#), [Y](#)

## Y -- "yes" icon

- Current value: Y = 
- Related: [H](#), [I](#), [ICON](#), [M](#), [N](#), [P](#), [Q](#), [S](#), [T](#), [U](#), [X](#)


## YELLOW -- start yellow colored text

- `YELLOW` is one of the shortcut macros predefined in [DefaultPreferences](#). See the section [shortcut macros](#) in that topic for a complete list of colors.

### Example:

```
%YELLOW% yellow text %ENDCOLOR%
```

Expands to: yellow text












 `<color>` text must end with `%ENDCOLOR%`. If you want to switch from one color to another one you first need to end the active color with `%ENDCOLOR%`, e.g. write

```
%RED% some text %ENDCOLOR% %GREEN% more text %ENDCOLOR%
```

Related: [ENDCOLOR](#), [DefaultPreferences](#), [StandardColors](#)

## Shortcuts

The following macros are [preference settings](#) and are frequently used in topic content.

- `%BR%` - line break
- `%BULLET%` - bullet sign
- `%BB%` - line break and bullet combined
- `%BB2%` - indented line break and bullet
- `%RED% text %ENDCOLOR%` - colored text (also `%YELLOW%`, `%ORANGE%`, `%PINK%`, `%PURPLE%`, `%TEAL%`, `%NAVY%`, `%BLUE%`, `%AQUA%`, `%LIME%`, `%GREEN%`, `%OLIVE%`, `%MAROON%`, `%BROWN%`, `%BLACK%`, `%GRAY%`, `%SILVER%`, `%WHITE%`)
- `%H%` -  Help icon
- `%I%` -  Idea icon
- `%M%` -  Moved to icon
- `%N%` -  New icon
- `%P%` -  Refactor icon
- `%Q%` -  Question icon
- `%S%` -  Red star icon
- `%T%` -  Tip icon
- `%U%` -  Updated icon
- `%X%` -  Alert icon
- `%Y%` -  Done icon

See [ShortcutMacros](#) for a full list of predefined shortcuts.

---

**Related Topics:** [UserDocumentationCategory](#)

--- This topic: System > WebHome > Macros

Topic revision: 08 Oct 2011, ProjectContributor

Copyright © by the contributing authors. All material on this site is the property of the contributing authors.

Ideas, requests, problems regarding Foswiki? [Send feedback](#)

