

Table of Contents

Main Web Preferences.....	1
Appearance.....	1
Wiki application settings.....	1
Access control.....	1
Tools.....	1
Preference Settings.....	1
Writing preference settings.....	2
Hiding preference settings.....	3
Order of preference settings.....	3
Preference settings and topic revision history.....	4
Parameters.....	4
Parameter defaults.....	4
Access Control Settings.....	4
Local values for preferences.....	4
Related Topics.....	5

Main Web Preferences

Appearance

- Set WEBBGCOLOR = #FFEFA6
 - ◆ web-specific background color, current color
- Set SITEMAPLIST = on
- Set WEBSUMMARY = Welcome to GSI Wiki ... meet people on this site

Wiki application settings

- WebTopicEditTemplate
 - ◆ default template for new topics in this web, if this topic does not exist, site-level default template will be used
- Set WEBFORMS = UserForm
 - ◆ Comma separated list of forms that can be attached to topics in this web. See DataForms for more information.

Access control

Users or groups who are allowed to view, change and rename topics in the Main web. Remember that an empty setting is a valid setting; setting DENYWEBVIEW to nothing means that anyone can view the web. If you intend to lock down the Main to only allow registration, you need to add the built in `RegistrationAgent` to the `ALLOWWEBCHANGE` setting.

- Set DENYWEBVIEW =
- Set ALLOWWEBVIEW =
- Set DENYWEBCHANGE =
- Set ALLOWWEBCHANGE = AdminGroup, RegistrationAgent
- Set DENYWEBRENAME =
- Set ALLOWWEBRENAME = AdminGroup

Users or groups allowed to change or rename this WebPreferences topic: (e.g., AdminGroup)

- Set ALLOWTOPICCHANGE = AdminGroup
 - Set ALLOWTOPICRENAME = AdminGroup
-

Tools

- Rename, move or delete this web

Show help Hide help
#SettingPrefs

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
```

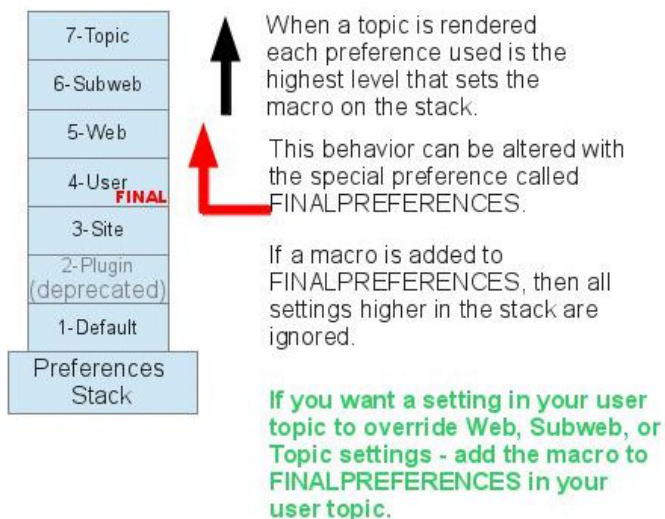
Macros defined using preference settings are expanded by enclosing their name in percent signs. So when you write %WEBBGCOLOR%, it gets expanded to #FFEEFA6

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. In (some) plugin documentation topics. (Deprecated)
3. SitePreferences
4. In user topics, if the user has one (yours is Main.WikiGuest)
5. WebPreferences
6. Sub-webs inherit the WebPreferences of their parent
7. In the topic being accessed

In this list, `Set` statements which occur at numerically higher 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.



Writing preference settings

Preference settings are written as a simple bullet. In TML, they are written as 3-spaces, asterisk, equals, value

```
* Set MYSETTING = My setting value
```

When using the Wysiwyg editor, click the "Bullet" button and write the setting as a simple bullet. Don't include the asterisk.

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
```

Preference settings are **case sensitive**. (Foswiki by convention always writes settings in upper case.)

```
* Set lower = This is LOWER
* Set LOWER = This is UPPER
* Set System.LoWeR = This is MIXED
Expand %lower%, %LOWER% and %LoWeR%
```

Expand %lower%, %LOWER% and %LoWeR%.

i preference settings can easily be disabled with a # sign. Example:

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

Hiding preference settings

i 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
-->
```

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.

i **Caution** If your topic will be used in an `INCLUDE`, it is recommended to not use HTML comments. Instead, set preferences into the topic metadata by using the "Edit Settings for this topic" button on the "More topic actions" page. Settings in an included topic are always ignored, but nested comments will break the HTML.

Order of preference settings

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.

Preference settings and topic revision history

Foswiki always reads the settings 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

Parameters

Note that %CONDITIONS% expands differently when this example is viewed in Macros. This is because Set statement 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 [[System.%BASETOPIC%][%BASETOPIC%]] the %WHAT% is %STATE% today
```

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

- %CONDITIONS{WHAT="sea" STATE="choppy"}%
 - ◆ expands to %CONDITIONS{WHAT="sea" STATE="choppy"}%.

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 %WEATHER%
- %WEATHER{"sunny"}% expands to %WEATHER{"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.

Deprecation warning. The setting used in this example, `EDITBOXHEIGHT`, is being deprecated and will be removed from Foswiki 1.2. Note that if the edit box size is changed using the javascript controls in the lower right corner of the edit box window, those settings will be used, and the `EDITBOX*` settings will be ignored.

Related Topics

- `System.DefaultPreferences`, `Main.SitePreferences` - site-level preferences
- `WikiUsers` - list of user topics. User topics can have optional user preferences
- `Macros` - what macros are and how to use them, including a list of common macros
- `AccessControl` - explains how to restrict access by users or groups

This topic: [Main](#) > [WebHome](#) > [WebPreferences](#)

Topic revision: 2014-02-11, [AdminUser](#)



Copyright &© by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

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