

Table of Contents

Search Pattern Cookbook.....	1
Pattern 1: Extract values from a table.....	1
Problem definition.....	1
Solution.....	1
Pattern 2: List generated from form classification.....	1
Problem.....	1
Test case.....	1
Solution.....	2
Pattern 3: Creating lists of TWiki usernames.....	2
Problem.....	2
Solution 1: Appropriate for Sep 2004 TWiki (Cairo).....	2
Solution 2: As Solution 1, but with possibility for multi-selecting usernames.....	2
Solution 3: Appropriate for TWiki 4 (Dakar).....	3
Pattern 4: Extract the parent of a given topic.....	3
Problem.....	3
Solution 1: Using META.....	3
Solution 2: Using SpreadSheetPlugin.....	3
Test Case.....	4
Solution 3: Using IF statement.....	4
Test Case.....	4
Pattern 5: Show all Children of a given topic.....	4
Problem.....	4
Solution.....	4
Pattern 6: Search and display the home topics of public webs in a list.....	4
Problem.....	4
Solution.....	4
Test case.....	5
Alternative solution.....	5
Pattern 7: Create a select box with values from a bullet list.....	5
Problem.....	5
Solution.....	5
Pattern 8: Extract a value from a named bullet list item.....	5
Problem.....	5
Solution.....	5
Test case.....	6
Pattern 9: Search for Form and Meta data: explained.....	6
Problem.....	6
Solution.....	6
Pattern 10: Search all topics that have been moved.....	7
Problem.....	7
Solution.....	7

Search Pattern Cookbook

The Search function in TWiki is very powerful. Especially searches using a RegularExpression play an important part of tapping TWiki's full potential. Unfortunately RegularExpressions can be incredibly obscure to the uninitiated.

Most people not familiar (enough) with Regular Expressions mostly cut and paste (and maybe tweak) from existing examples. This page intends to collect lots of examples together.

Pattern 1: Extract values from a table

Problem definition

Suppose there is a topic with a table defining entries in a TWikiForm. I.e. they define select menu items in a form template. They are then formatted like:

```
| *Name* | *Type* | *Tooltip message* |  
| option1 | option | |  
| option2 | option | |  
| option3 | option | |
```

How to extract the 'name' values, i.e. 'option1', 'option2' and 'option3' and put them in a HTML form select input?

Solution

The following search pattern can be employed:

```
<form>  
<select>  
%SEARCH{ "^\|[\^|]*\| *option *\|" topic="%TOPIC%" type="regex" multiple="on" nosearch="on" notot  
</select>  
</form>
```

which is, in effect:

Pattern 2: List generated from form classification

Problem

Imagine a TWiki form-based topic classification, i.e. every page has a form with several fields. How to:

1. create a search to display all topics where one form field is set to a certain value
2. create a search to filter the list above based on the values of a second form field

Test case

In practice:

Image a TWiki form with two fields:

- TopicClassification = One, Two or Three


```
<select name="topic" size="2" MULTIPLE>
```

Please note that the Search pattern is unchanged compared to Solution 1. The change is in the HTML form element.

The abovementioned modification is, in effect:

Solution 3: Appropriate for TWiki 4 (Dakar)

When the User information is stored in a UserForm (as is default in Dakar) then this list can be generated as follows:

```
<form name="testing" action="%SCRIPTURLPATH{"view"}%/%USERSWEB%" method="get">
<select name="topic">
<option>Select user...</option>
%SEARCH{"%META:FORM.*[U]serForm" web="%USERSWEB%" type="regex" casesensitive="on" nosearch="on" f
</select>
<input type="submit" value="Go" />
</form>
```

In the above example:

- **META:FORM.*[U]serForm** will search for all topics with a UserForm attached - change this if you have a different form where userdata is stored. Please note that this search does not actually extract anything from the form - it just uses it to identify the appropriate pages
- **excludetopic="Test*, TWiki*"** allows to skip all topics starting with Test and TWiki, such as TestUser or TWikiAdmin. Use this if you have any special users who you do not want appearing in this list

Pattern 4: Extract the parent of a given topic

Problem

How to get to the parent of the current topic to display on the page?

Solution 1: Using META

Since TWiki 4.0 you can now use the META variable:

```
%META{ "parent" dontrecurse="on" }%
```

Solution 2: Using SpreadSheetPlugin

You might think that the following Search would do the trick:

```
%SEARCH{ "^%BASETOPIC%" scope="topic" nosearch="on" type="regex" nonoise="on" format=" * $parent
```

However, the `$parent` link fails if the topic has no parent set (`$parent` will be empty). You can use some TWiki:Plugins/SpreadSheetPlugin magic to conditionally link to the parent or to WebHome:

```
$percentCALC{ $IF( $EXACT( $parent, ), <nop>, $NOP( * $parent ) ) }$percent
```

So the total Search query to find a topic's parent topic is:

```
%SEARCH{ "^%BASETOPIC%" scope="topic" type="regex" nonoise="on" format="$percentCALC{$IF($EXACT($
```

Test Case

The parent topic of this topic is:

- FormattedSearch

Solution 3: Using IF statement

This pattern can be rewritten using %IF%, removing the dependency on SpreadSheetPlugin:

```
%SEARCH{ "%BASETOPIC%" web="%BASEWEB%" scope="topic" type="regex" nonoise="on" format="$percentI
```

Test Case

The parent topic of this topic is:

Pattern 5: Show all Children of a given topic

Problem

How to get to the list of all children of the current topic to display on the page?

Solution

The parent information is stored in the META:TOPICPARENT meta data. Do a SEARCH to find all topic parent meta data pointing to the current topic:

```
Children:
%SEARCH{ "META\:TOPICPARENT.*\"%TOPIC%\"" type="regex" nonoise="on" format="[$topic]" separator
```

Note: Replace %TOPIC% with %BASETOPIC% if you put this SEARCH into the skin or a sidebar.

Pattern 6: Search and display the home topics of public webs in a list

Problem

How to find and display public webs in a drop down list box.

Solution


Thanks to TWiki:Main.PeterThoeny for these solutions.

```
<form>
<select name="topic">
<option value="%TOPIC%">Select...</option>
%SEARCH{ "%HOMETOPIC%" scope="topic" web="all" topic="%HOMETOPIC%" format="<option value=\" $web.$
</select>
<input type="submit" value="Go" />
</form>
```

Test case

Public webs of TWiki.

Number of topics: 1

 For private webs, or any other webs you wish to exclude from the display, use "on" for the `Exclude web from a web="all"` search setting in the relevant web's `WebPreferences` topic.

Alternative solution

This result can also be accomplished with the `%WEBLIST%` variable.

Pattern 7: Create a select box with values from a bullet list

Problem

We have a topic with a bullet list with category names. In another topic we want to offer these values in a select box dropdown.

For example, `CategoryList` has:

- Clients
- People
- Rooms
- Buildings

Solution

The following search pattern can be employed:

```
<select name="type">
<option>Select category...</option>
%SEARCH{" * \s*.*?" topic="CategoryList" type="regex" multiple="on" casesensitive="on" nosummary
</select>
```

To render the bullet list as a comma-separated list, use the `separator` parameter:

```
%SEARCH{" * \s*.*?" topic="CategoryList" type="regex" multiple="on" casesensitive="on" nosummary
```

Pattern 8: Extract a value from a named bullet list item

Problem

Display the user name in the user's topic title

Solution

Search for the `Name`: entry.

```
%SEARCH{" * [N]ame: " topic="%TOPIC%" type="regex" casesensitive="on" nosummary="on" nosearch="
```

Test case

Test case

To create a test case, we will put a name entry here:

- Name: John Doe

Search result:

Pattern 9: Search for Form and Meta data: explained

Problem

Below is an example of a search that searches form data. The questions are:

- why is this searching the metadata, shouldn't it just search the text?
- what is the meaning of the `td..td` in the search expression?

```
%SEARCH{ "[S]tatus.*(td..td|value\=).*[W]aiting" casesensitive="on" type="regex"
nosearch="on" nototal="on" format="| [[${topic}]<br /> ($date - $rev -
[[%SCRIPTURLPATH{rdiff}%/$web/${topic} [Diffs]]) |"]%
```

Solution

`%SEARCH` depends on `grep`, and `grep` searches the whole file, including the meta data.

An example meta data form field is:

So a search for a form field could look like:

```
%SEARCH{ "[O]peratingSystem.*value\=.*[O]sWin" type="regex" ... }%
```

- Using square brackets is a trick to avoid a hit on the topic doing the search.
- The `.*` indicate that there can be any number of characters between `OperatingSystem` and `value` in the (whole) file

Now the original file format of the category table (the predecessor of the TWiki forms) looks like this:

```
<td valign="top" align="right"> OperatingSystem: </td><td> OsWin </td>
```

The following search finds topics in the old and new format:

```
%SEARCH{ "[O]peratingSystem.*(td..td|value\=).*[O]sWin" type="regex" ... }%
```

The `td..td` matches `td<>td`; a simple search on `"[O]peratingSystem.*[O]sWin"` could find a hit in the topic text by coincidence.

A simple `%SEARCH{ "[O]peratingSystem.*value\=.*[O]sWin" ... }%` search is sufficient if you do not have topics in the old format.

Pattern 10: Search all topics that have been moved

Problem

How would I go about listing all moved topics ?

Solution

Search for the META:TOPICMOVED meta data. Type this:

```
Moved topics: %SEARCH{ "%META\ :TOPICMOVED" type="regex" format="$topic, " nosearch="on"
noheader="on" nosummary="on" }%
```

to get this (limited to 10 results):

Moved topics: BibtexSearchTemplate, DeleteRevisionsAddOn, LdapUserInfoBox, SearchResults,

Number of topics: 4

Related Topics: UserDocumentationCategory, SearchHelp, TWikiVariables#VarSEARCH, FormattedSearch, RegularExpression

-- **Contributors:** [TWiki:Main.AntonAylward](#), [TWiki:Main.ArthurClemens](#), [TWiki:Main.JosMaccabiani](#), [TWiki:Main.PeterThoeny](#), [TWiki:Main.SueLocke](#)

This topic: TWiki21Nov > SearchPatternCookbook

Topic revision: r6 - 2009-04-24 - TWikiContributor



Copyright &© 2008-2022 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

or Ideas, requests, problems regarding TWiki? use [Discourse](#) or [Send feedback](#)