See also Front-ends to Tracker data

Plugin List
If you are searching for the list of all plugins, see All Plugins. A graphical user interface (ListGUI) was added in Tiki18 and massively improved in Tiki19. Starting in Tiki21, it is has been enhanced to use with Federated Search.

Introductory Remarks

**LIST** is a very powerful and flexible wiki plugin that can return and output data (a listing) of information using various sorts, filters, etc. It uses the search data provided by the improved search index so it should be emphasized that this means that only the data that has actually been indexed by the Search and List from Unified Index can be accessed. Good knowledge of how Unified Index works is therefore recommended.

By utilizing a full-text search across most major Tiki features, which is then enhanced by a programmable filter, OUTPUT, DISPLAY, and SORT control blocks, this combination can render almost any information in the database in any format desired. This means that LIST is similar in some respects to Pretty Trackers output of TrackerList plugin but it is not limited to just the Tracker data. When combined with the PluginCustomSearch plugin, LIST can replace TrackerList and TrackerFilter plugin usage and can considerably extend it!! Please also see PluginListExecute.

Please note you can use nested **LIST**.

Control blocks vs. Wiki Plugins

"Control blocks" are quite similar to "wiki plugins", as they use the very same syntax of parameters and values. The positions of the parameters can be switched around in wiki plugins as well as in List control blocks. Both allow to leave out the quotation marks around the values of the parameters when the values contain no empty spaces.

However, for both, control blocks and wiki plugins, it makes sense to stick to some best practices in respect of consistent sort order. This is not mandatory but makes it easier to understand the principle, to read the own code, and to work together in teams. This is especially valid for the LIST control blocks, as there are usually a lot of control blocks used in one LIST plugin.

The difference between "control blocks" and "wiki plugins" is, that you cannot use a "control block" stand-alone outside one of the plugins, which use the "PluginList" syntax, as there are "PluginList", "PluginCustomSearch" and "PluginListExecute" (?). If you place a single List "control block" stand alone on a wiki page, either nothing happens or you'll get an error message.

"Wiki plugins", instead, can be used stand-alone anyplace where you can use wiki syntax and, to some extent, you can nest them as well.

Clarifying a potential confusion

Potentially confusing is the difference between a control block and a parameter both called "format".

The parameter "format" (in lower case) inside the control block "{display}" is responsible for how the displayed result is rendered. Maybe it also could have been named "render" instead of "format", but "format" was the naming decision of our coders and that makes sense as well, maybe even better sense for some reason.

The control block "{FORMAT(name=...)}" (in upper case) wraps around the control block "{display}" and
is for example responsible for the reference to the control block column (when we consider the example of a tracker table display).

The other potentially confusing fact is, is the difference and same time similarity of the parameters "field" and "name", where we use always "field" in the {filter} and in the {column} inside the "{OUTPUT(template=table))" control block. Contrarily we use "name" in the optional "{display} and {FORMAT()}" control blocks.

**When we use this ...**

```plaintext
{filter field="tracker_id" content="10"} {OUTPUT(template="table"}) {column
field="tracker_field_permanent_name_1" label="columntitle" mode="raw"} {column
field="tracker_field_permanent_name_2" label="columntitle" mode="raw"} {OUTPUT}
```

... then the "field" parameter contains a "Unified Index field" - in this case a tracker field which we want to display.

**But when we use this ...**

```plaintext
{filter field="tracker_id" content="10"} {OUTPUT(template="table"}) {column
field="reference_1" label="columntitle" mode="raw"} {column
field="reference_2" label="columntitle" mode="raw"} {OUTPUT}
{FORMAT(name="reference_1")}{display name="tracker_field_permanent_name_1"
format=trackerrender editable=inline default="n.a."} {FORMAT}
{FORMAT(name="reference_2")}{display name="tracker_field_permanent_name_2"
format=trackerrender editable=inline default="n.a."} {FORMAT}
```

... we referenced (passed) the actual value of the "field" parameter of control block "{column ...}" to the subcontrol block "{display ...}" inside the control block "{FORMAT(...)", where we for some reason cannot use a parameter "field".

When we reference the original content of "field" from one to another control block, both control blocks need to "know" about each other, which you see as obvious, when you have more than only one column. So the content of the column's field parameter will be replaced with a reference string and this reference string has to be repeated in the FORMAT's name parameter. Now the column control block and the FORMAT control block are interlinked with each other. Finally, the original content of the columns field parameter (which is a "Unified Index field") has to be placed into the display's name parameter.

**In other words:**

"Unified Index field string" goes from "column field" to "display name"

And a "reference string" is added to "column field" and "FORMAT name"

Important to know is the content of the page [Search and List from Unified Index](#), where you find a list of all available "Unified Index fields" for which you can filter and in respect of tracker based tables you can use to create columns.

Syntax Overview

The overall format is the same as any other plugin:

```plaintext
{LIST()} body content {LIST}
```
Any of the following control blocks with their own plugin-like syntax are placed in the body of the LIST plugin to define the search query that will be carried out and how the resulting list will be presented:

<table>
<thead>
<tr>
<th>plugin-like control block</th>
<th>description</th>
<th>version</th>
<th>default or required</th>
<th>see below for more detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>list or pagination</td>
<td>The LIST plugin will display 50 results by default but depending on the output you might want to decrease the visible amount of results to improve performance.</td>
<td>pagination from Tiki 11</td>
<td>default is 50 items</td>
<td>see the child page PluginList pagination or list control block for full details and worked examples</td>
</tr>
<tr>
<td>filter</td>
<td>is a required control block in the LIST body content and is used to define the search query that will be carried out ie what objects from the complete set that has been indexed by the Unified Search will be included in the resultant list</td>
<td>required</td>
<td></td>
<td>see the child page PluginList filter control block for full details and worked examples</td>
</tr>
<tr>
<td>Control Block</td>
<td>Description</td>
<td>Default or Required</td>
<td>Notes</td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
<td>---------------------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>OUTPUT</td>
<td>defines what the output 'template' will be by either using one of several standard/built-in templates eg table, medialist or carousel, or by referencing a user-defined wiki or smarty template. For the table template, for example, the body content is used to define which columns/fields are shown.</td>
<td>optional</td>
<td>see the child pages PluginList output control block and PluginList advanced output control block for full details and worked examples.</td>
<td></td>
</tr>
<tr>
<td>ALTERNATE</td>
<td>used in conjunction with the OUTPUT control block it can define an alternative output when an individual item (row) from a search/listing has no value.</td>
<td>optional</td>
<td>see the child page PluginList output control block for more details and worked examples.</td>
<td></td>
</tr>
<tr>
<td>FORMAT</td>
<td>The FORMAT control block is used to create individually templated objects that can then be used in any of the individual OUTPUT methods.</td>
<td>optional</td>
<td>see the child page PluginList format control block for full details and worked examples.</td>
<td></td>
</tr>
</tbody>
</table>
**Plugin-like control block**

<table>
<thead>
<tr>
<th>DISPLAY:</th>
<th>Used to define placement and formatting of individual objects</th>
<th>version</th>
<th>default or required</th>
<th>see below for more detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>SORT:</td>
<td>allows the resultant list of objects to be sorted in a specified order</td>
<td>version</td>
<td>default or required</td>
<td>see below for more detail</td>
</tr>
<tr>
<td>index</td>
<td>allows the inclusion of Federated Search results. You can further filter by the _index field using the filter control block</td>
<td>version</td>
<td>default or required</td>
<td>optional</td>
</tr>
</tbody>
</table>

**Parameters**

While Plugin List is generally configured with plugins (some say *control blocks*) contained in the body as described above, there are a few standard parameters like other plugins, they are:

Search for, list, and filter all types of items and display custom-formatted results.

*Introduced in Tiki 7.*

*Go to the source code*

*Preferences required:* wikiplugin_list, feature_search

<table>
<thead>
<tr>
<th>Parameter (body of plugin)</th>
<th>Accepted Values</th>
<th>Description</th>
<th>Default</th>
<th>Since</th>
</tr>
</thead>
<tbody>
<tr>
<td>cacheexpiry</td>
<td>word</td>
<td>Time before cache is expired in minutes.</td>
<td>20.0</td>
<td></td>
</tr>
<tr>
<td>multisearchid</td>
<td>text</td>
<td>This is for much better performance by doing one search for multiple LIST plugins together. Render results from previous {MULTISEARCH(id-x)}...{MULTISEARCH} block by providing the ID used in that block.</td>
<td>20.0</td>
<td></td>
</tr>
<tr>
<td>cache</td>
<td>y n a</td>
<td>Cache output of this list plugin.</td>
<td>20.0</td>
<td></td>
</tr>
</tbody>
</table>
Basic worked example

step by step instructions to be added here for a basic worked example

In the meantime, you can see a basic working example here: GeoLocation (or play in your tiki with a similar basic example after applying the profile Easy Geoblog, which is available at the Profiles Wizard).

Body content elements of the LIST Plugin

Each of the principle plugin-like control blocks is described in more detail in the following set of child pages:

- PluginList pagination or list control block
- PluginList filter control block
- PluginList output control block
- PluginList format control block
- PluginList display control block
- PluginList sort control block
- PluginList advanced output control block
- PluginList multisearch output control block
- PluginList aggregate control block
- PluginList overview about control blocks parameters and values
- Troubleshooting
- GUI
- Hacks and Fun
- PluginList and Metatags - SEO

Additional General Notes on the Syntax

- The field argument in the content filter can contain multiple fields separated by commas.
All the fields that are indexed can be referenced by the various plugin-like control blocks e.g. filter, FORMAT, etc and a complete list of fields for each object type can be found in the Search and List from Unified Index documentation.

More Worked Examples

Example Tracker item with Comments

Item

{{LIST}} {pagination max="1"} {filter type="trackeritem"} {filter field="object_id" content=""} {LIST}

Comments

{{LIST}} {filter type="comment"} {filter field="parent_object_id" content=""} {filter field="parent_object_type" content="trackeritem"} {LIST}

Example Tracker item with a picture (file)

{{LIST}} {filter type="trackeritem"} {OUTPUT} {display name="fathername"} %%%
 {display name="fatherpict"} %%% {display name="id"} {OUTPUT}
 {FORMAT(name="fathername")} {display name="tracker_field_fatherName"}{FORMAT}
 {FORMAT(name="fatherpict")} {display name="tracker_field_fatherPicture" format="trackerrender"}{FORMAT} {FILTER(name="id")}{display name="tracker_field_parentId"}{FILTER} {LIST}

Same Example as above within a table

(see: LIST - OUTPUT control block)

{{LIST}} {filter content="1" field="tracker_id"} {OUTPUT(template="table")} {column label="Father Name" field="fathername"} {column label="Picture" field="fatherpict" mode="raw"} {column label="ID" field="id"} {OUTPUT}
 {FORMAT(name="fathername")} {display name="tracker_field_fatherName"}{FORMAT}
 {FORMAT(name="fatherpict")} {display name="tracker_field_fatherPicture" format="trackerrender"}{FORMAT} {FILTER(name="id")}{display name="tracker_field_parentId"}{FILTER} {LIST}

Example Blog Post List

This code:

{{LIST}} {list max="10"} {filter type="blog post"} {filter content="1" field="blog_id"} {LIST}

Would produce on this site:

- Protect all sessions with HTTPS enabled
- Check-file-indexing is lost in space?
- Comments enabled for registered users
- Packages
- Preference feature jquery validation disabled (Preference name: feature_jquery_validation)
- Copyright turned off
- Disabled "Users can choose to stay in SSL mode after an HTTPS login" to have SSL login always on.
• DocContributors granted the perms to edit doc.tw.o homepage
• Wiki Structures re-enabled
• site upgraded to the released tiki 2.0 stable

Tutorials

• Tutorial - Display Tracker Data with Plugin List
• LIST plugin demo and nesting tracker items - Youtube

Additional child pages of the LIST documentation

• PluginList pagination or list control block
• PluginList filter control block
• PluginList output control block
• PluginList format control block
• PluginList display control block
• PluginList sort control block
• PluginList advanced output control block
• PluginList multisearch output control block
• PluginList aggregate control block
• PluginList overview about control blocks parameters and values
• Troubleshooting
• GUI
• Hacks and Fun
• PluginList and Metatags - SEO

Related

See also:

• PluginTrackerList
• PluginTrackerList To PluginList Converter
• PluginTracker

Aliases

• Plugin List