Use this wiki plugin to generate forms from a tracker. Currently only able to list data, the TrackerQueryTemplate plugin simply obtains data from a tracker and allows an editor the ability to list the tracker data as he sees fit.

**Parameters**

*Introduced in Tiki 10. Required parameters are in bold. Preferences required: feature_trackers, wikiplugin_trackerquerytemplate*

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Accepted Values</th>
<th>Description</th>
<th>Default</th>
<th>Since</th>
</tr>
</thead>
<tbody>
<tr>
<td>(body of plugin)</td>
<td></td>
<td>Wiki Syntax, with variables from tracker query.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>tracker</strong></td>
<td>text</td>
<td>The name of the tracker to be queried, or if byname=&quot;n&quot;, the tracker ID.</td>
<td>10.0</td>
<td></td>
</tr>
<tr>
<td><strong>andfilters</strong></td>
<td>text</td>
<td>Apply &quot;and&quot; filters to fields. Format: field:value;field:value;field:value, where field may be the field name or ID depending on the setting for the byname parameter.</td>
<td>10.0</td>
<td></td>
</tr>
<tr>
<td><strong>likefilters</strong></td>
<td>text</td>
<td>Apply &quot;like&quot; filters to fields. Format: field:value;field:value;field:value, where field may be the field name or ID depending on the setting for the byname parameter.</td>
<td>10.0</td>
<td></td>
</tr>
<tr>
<td><strong>orfilters</strong></td>
<td>text</td>
<td>Apply &quot;or&quot; filters to fields. Format: field:value;field:value;field:value, where field may be the field name or ID depending on the setting for the byname parameter.</td>
<td>10.0</td>
<td></td>
</tr>
<tr>
<td><strong>debug</strong></td>
<td>(blank)</td>
<td>Turn tracker query debug on (off by default).</td>
<td></td>
<td>10.0</td>
</tr>
<tr>
<td><strong>getlast</strong></td>
<td>(blank)</td>
<td>Retrieve only the last item from the tracker.</td>
<td></td>
<td>10.0</td>
</tr>
<tr>
<td><strong>render</strong></td>
<td>(blank)</td>
<td>Render as needed for trackers (default).</td>
<td>y</td>
<td>10.0</td>
</tr>
<tr>
<td>Parameter</td>
<td>Type</td>
<td>Description</td>
<td>Value</td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>------</td>
<td>-------------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>byname</td>
<td>(blank)</td>
<td>Use the tracker name instead of tracker ID in the <code>tracker</code> parameter. Also use the field name instead of field ID in the filter parameters. Set to Yes (y) to use names (default) or No (n) to use IDs.</td>
<td>y 10.0</td>
<td></td>
</tr>
<tr>
<td>itemid</td>
<td>digits</td>
<td>Item id of tracker item</td>
<td>10.0</td>
<td></td>
</tr>
<tr>
<td>itemids</td>
<td>digits</td>
<td>Item id of tracker items, separated with comma</td>
<td>11.0</td>
<td></td>
</tr>
</tbody>
</table>

**Features**

- Supports nesting, where values in settings are from the parent tracker, and those in the body are from the current tracker.
- When using `byname="y"` (default), variables are accessed `$field name$` (rendered) and `$~field name$` (unrendered).
- When `byname="n"`, variables are `{f_id}` (rendered) and `{~f_id}` (unrendered).
- Filters are in the format `field:value;field:value;field:value;`.

**Simple Example**

Here the:

- tracker id = 1
- username field id = 1
- email field id = 2

```
{TRACKERQUERYTEMPLATE(tracker="1" orfilters="1:sam;1:bob;")}
Username: {f_1}
Email: {f_2}
{TRACKERQUERYTEMPLATE}
```

**Nested Example**

Here we use the `byname` parameter, which can be more intuitive. **NOTICE** how `$~username$` comes from tracker named "User Tracker" and not "User Sports Tracker", thus aiding in complex joining scenarios.

```
{TRACKERQUERYTEMPLATE(tracker="User Tracker" byname="y" orfilters="username:sam;username:bob;")}
Username: $username$
Email: $email$
Sports:
    {TRACKERQUERYTEMPLATE(tracker="User Sports Tracker" byname="y" orfilters="username:$~username$")}
    *$sport$
{TRACKERQUERYTEMPLATE}
```
What is nesting?

Nesting is when an object is embedded in the same type of object, a visual:

- root
  - nested in root
    - nested in previous nested

or in the case of TrackerQueryTemplate nested 2 times:

```
{TRACKERQUERYTEMPLATE()}
{TRACKERQUERYTEMPLATE()}
{TRACKERQUERYTEMPLATE()}{TRACKERQUERYTEMPLATE}
{TRACKERQUERYTEMPLATE}
{TRACKERQUERYTEMPLATE}
```