

# TicketQuery Wiki Macro

The TicketQuery macro lets you display information on tickets within wiki pages. The query language used by the `[[TicketQuery]]` macro is described in [TracQuery](#) page.

## Usage

### `[[TicketQuery]]`

Wiki macro listing tickets that match certain criteria.

This macro accepts a comma-separated list of keyed parameters, in the form "key=value".

If the key is the name of a field, the value must use the syntax of a filter specifier as defined in [TracQuery#QueryLanguage](#). Note that this is *not* the same as the simplified URL syntax used for `query:` links starting with a `?` character. Commas (`,`) can be included in field values by escaping them with a backslash (`\`).

Groups of field constraints to be OR-ed together can be separated by a literal `or` argument.

In addition to filters, several other named parameters can be used to control how the results are presented. All of them are optional.

The `format` parameter determines how the list of tickets is presented:

- **list** -- the default presentation is to list the ticket ID next to the summary, with each ticket on a separate line.
- **compact** -- the tickets are presented as a comma-separated list of ticket IDs.
- **count** -- only the count of matching tickets is displayed
- **rawcount** -- only the count of matching tickets is displayed, not even with a link to the corresponding query (*since 1.1.1*)
- **table** -- a view similar to the custom query view (but without the controls)
- **progress** -- a view similar to the milestone progress bars

The `max` parameter can be used to limit the number of tickets shown (defaults to **0**, i.e. no maximum).

The `order` parameter sets the field used for ordering tickets (defaults to **id**).

The `desc` parameter indicates whether the order of the tickets should be reversed (defaults to **false**).

The `group` parameter sets the field used for grouping tickets (defaults to not being set).

The `groupdesc` parameter indicates whether the natural display order of the groups should be reversed (defaults to **false**).

The `verbose` parameter can be set to a true value in order to get the description for the listed tickets. For **table** format only. *deprecated in favor of the `rows` parameter*

The `rows` parameter can be used to specify which field(s) should be viewed as a row, e.g.  
`rows=description|summary`

The `col` parameter can be used to specify which fields should be viewed as columns. For **table** format only.

For compatibility with Trac 0.10, if there's a last positional parameter given to the macro, it will be used to specify the `format`. Also, using "&" as a field separator still works (except for `order`) but is deprecated.

## Example

Example	Result	Macro
Number of <b><u>Triage tickets:</u></b>	<b>0</b>	<code>[[TicketQuery(status=new&amp;milestone=, count)]]</code>
Number of <b>new tickets:</b>	<b>0</b>	<code>[[TicketQuery(status=new, count)]]</code>
Number of <b>reopened tickets:</b>	<b>0</b>	<code>[[TicketQuery(status=reopened, count)]]</code>
Number of <b>assigned tickets:</b>	<b>0</b>	<code>[[TicketQuery(status=assigned, count)]]</code>
Number of <b>invalid tickets:</b>	<b>0</b>	<code>[[TicketQuery(status=closed, resolution=invalid, count)]]</code>
Number of <b>workforme tickets:</b>	<b>0</b>	<code>[[TicketQuery(status=closed, resolution=workforme, count)]]</code>
Number of <b>duplicate tickets:</b>	<b>0</b>	<code>[[TicketQuery(status=closed, resolution=duplicate, count)]]</code>
Number of <b>wontfix tickets:</b>	<b>0</b>	<code>[[TicketQuery(status=closed, resolution=wontfix, count)]]</code>
Number of <b>fixed tickets:</b>	<b>0</b>	<code>[[TicketQuery(status=closed, resolution=fixed, count)]]</code>
<b>Total number of tickets:</b>	<b>0</b>	<code>[[TicketQuery(count)]]</code>
Number of <b>tickets reported or owned by current user:</b>	<b>0</b>	<code>[[TicketQuery(reporter=\$USER, or, owner=\$USER, count)]]</code>
	<b>0</b>	<code>[[TicketQuery(created=thismonth.., count)]]</code>

<b>Example</b>	<b>Result</b>	<b>Macro</b>
Number of tickets created this month:		
Last 3 modified tickets:	No results	<code>[[TicketQuery(max=3,order=modified,desc=1,compact)]]</code>
Details of ticket #1:		<code>[[TicketQuery(id=1,col=id owner reporter,rows=summary,table)]]</code>
	<b><u>Ticket</u> <u>Owner</u> <u>Reporter</u></b>	
	No tickets found	

## Using the `[[TicketQuery]]` Macro

The `?TicketQuery` macro lets you display lists of tickets matching certain criteria anywhere you can use [WikiFormatting](#).

Example:

```
[[TicketQuery(version=0.6|0.7&resolution=duplicate)]]
```

This is displayed as:

No results

Just like the [query: wiki links](#), the parameter of this macro expects a query string formatted according to the rules of the simple [ticket query language](#). This also displays the link and description of a single ticket:

```
[[TicketQuery(id=123)]]
```

This is displayed as:

No results

A more compact representation without the ticket summaries is:

```
[[TicketQuery(version=0.6|0.7&resolution=duplicate, compact)]]
```

This is displayed as:

No results

If you wish to receive only the number of defects that match the query, use the `count` parameter:

```
[[TicketQuery(version=0.6|0.7&resolution=duplicate, count)]]
```

This is displayed as:

## Q

A graphical use of the macro is with the `format=progress` attribute:

```
[[TicketQuery(milestone=0.12.8&group=type,format=progress)]]
```

For example for one of the upcoming milestones, bars are shown by ticket type:

---

See also: [TracQuery](#), [TracTickets](#), [TracReports](#), [TracGuide](#)