

Query option, The sql for field to allow groups option.

2022/05/09 17:40 - Admin Redmine

<b>ステータス:</b>	New	<b>開始日:</b>	2022/05/09
<b>優先度:</b>	高め	<b>期日:</b>	
<b>担当者:</b>		<b>進捗率:</b>	0%
<b>カテゴリ:</b>		<b>予定工数:</b>	0.00時間
<b>対象バージョン:</b>		<b>作業時間:</b>	0.00時間
Redmineorg_URL:	https://www.redmine.org/issues/20124	status_id:	1
category_id:	0	tracker_id:	3
version_id:	0	plus1:	0
issue_org_id:	20124	affected_version:	
author_id:	84718	closed_on:	
assigned_to_id:	0	affected_version_id:	
comments:	6		
<b>説明</b>			
<p>The simple commit help abundantly and allow us to grouping the custom queries.            When the available filters is added with the field name, the check already exists for sql field statement.            Another check is added when the field is added with grouping.</p> <p>Please update this patch in the core.</p> <p>Thanks.</p>			
<p>journals</p> <p>The advantage of this patch, is that I do not need to alias the sql_for_field method in my plugin!            I am also open for more elegant solution.</p> <p>Thanks.</p> <p>Where is @\$2@ from?</p> <p>@field =~ /^+./@ returns only @\$1@.</p> <p>For the below cases, With grouping of filters,</p> <p>Suppose, If I add a custom filter through a plugin patch</p> <pre>           add_available_filter("question.qassigned_to_id", :type =&gt; :list, :values =&gt; author_values,             :caption =&gt; :field_question_assigned_to_id, :name =&gt; I(:field_question_assigned_to_id_esi)           )           </pre> <p>The only way to handle the corresponding SQL query is to alias the sql_for_field.            Instead if the attached patch is provided,            I can avoid the alias and also have my method,</p> <pre>           def sql_for_qassigned_to_id_field(.....)             .... sql logic...           end           </pre> <p>Thank You.</p> <p>In your attachment:sql_for_fields_with_group-git.diff</p> <pre>           elsif field =~ /^+(.)\/&amp;&amp; respond_to?("sql_for_#{2}_field")             # specific statement           </pre>			

```
filters_clauses << send("sql_for_#{ $2 }_field", field, operator, v)
```

<http://ruby-doc.org/core-2.1.1/Regexp.html#class-Regexp-label-Special+global+variables>

\$1, \$2 and so on contain text matching first, second, etc capture group;

But your `@field =~ /\.+\/@` has only one parenthesis.

Yes, Realized my mistake. Thank You.  
Here is the Updated patch.

Thanks.

---