Redmine - Defect #32404
Group results is different between MySQL and PostgreSQL
2019-11-04 15:56 - Simon Hori

<table>
<thead>
<tr>
<th>Status:</th>
<th>New</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
</tr>
<tr>
<td>Category:</td>
<td>Issues filter</td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
</tr>
<tr>
<td>Resolution:</td>
<td></td>
</tr>
<tr>
<td>Start date:</td>
<td></td>
</tr>
<tr>
<td>Due date:</td>
<td></td>
</tr>
<tr>
<td>% Done:</td>
<td>0%</td>
</tr>
<tr>
<td>Estimated time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td>Affected version:</td>
<td></td>
</tr>
</tbody>
</table>

**Description**

Recently we’ve migrated a huge DB from MySQL to PostgreSQL. And noticed a small difference with Group results by function.

As you see the screenshot below, MySQL shows (blank) group first but PostgreSQL shows (blank) group last.

PostgreSQL-MySQL-NUL.png

I guess it is because `NULLS LAST` is the default for ASC order in PostgreSQL.
Can you enforce `NULLS FIRST` in the SQL query to postgres?

https://www.postgresql.org/docs/8.3/queries-order.html

We believe that it would be very much appreciated since many of our users want to group issues by assignee to find unassigned issues. Also it would be nice if Redmine works consistently irrespective of db engines.

Redmine version 3.4.11.stable
Ruby version 2.4.6-p354 (2019-04-01) [x86_64-linux]
Rails version 4.2.11.1
Database adapter PostgreSQL

Thank you.

**Related issues:**
Related to Redmine - Patch # 25066: Sort order with null placement (NULLS FIR...

**History**

#1 - 2019-11-04 16:00 - Simon Hori

I found a patch #25066 which is for a similar point but it is doing more than fixing the diff.

#2 - 2019-11-04 16:08 - Marius BALTEANU

- Related to Patch #25066: Sort order with null placement (NULLS FIRST, NULLS LAST) added

#3 - 2019-11-19 09:41 - Yuichi HARADA

https://www.redmine.org/builds/logs/build_trunk_postgresql_ruby-2.6_803.html

I think the reason why the test failed with TimelogControllerTest#test_index_grouped_by_issue is the same.

**Files**

PostgreSQL-MySQL-NUL.png 94.8 KB 2019-11-04 Simon Hori

2020-06-13 1/1