

## Redmine - Feature #33126

### Support custom fields when exporting users to CSV

2020-03-11 09:21 - Mizuki ISHIKAWA

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Go MAEDA	<b>% Done:</b>	0%
<b>Category:</b>	Accounts / authentication	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	4.2.0		
<b>Resolution:</b>	Fixed		
<b>Description</b> You can export the users list csv( <a href="#">#5957</a> ), but it does not include the value of UserCustomField. The value of UserCustomField should also be output as csv.			
<b>Related issues:</b> Related to Redmine - Patch #5957: Export users list to CSV <b>Closed</b> <b>2010-07-25</b>			

#### Associated revisions

##### Revision 19642 - 2020-04-04 03:42 - Go MAEDA

Support custom fields when exporting users to CSV (#33126).

Patch by Mizuki ISHIKAWA.

##### Revision 19643 - 2020-04-05 06:29 - Go MAEDA

RuboCop: Add "EnforcedStyle: sprintf" (#33126).

##### Revision 19767 - 2020-05-09 02:32 - Go MAEDA

Fix that custom fields in users CSV are not ordered by position value (#33126).

#### History

##### #1 - 2020-03-11 09:22 - Mizuki ISHIKAWA

- File *feature-33126.patch* added

I have attached a patch.

##### #2 - 2020-03-11 14:27 - Go MAEDA

- Target version set to Candidate for next major release

##### #3 - 2020-04-01 17:44 - Go MAEDA

- Target version changed from Candidate for next major release to 4.2.0

Setting the target version to 4.2.0.

##### #4 - 2020-04-04 03:42 - Go MAEDA

- Subject changed from Add UserCustomField value to csv of users list to Support custom fields when exporting users to CSV

- Status changed from New to Closed

- Assignee set to Go MAEDA

- Resolution set to Fixed

Committed the patch. Thank you for improving Redmine.

##### #5 - 2020-04-04 03:49 - Go MAEDA

- Related to Patch #5957: Export users list to CSV added

##### #6 - 2020-04-04 15:00 - Marius BĂLTEANU

- Status changed from Closed to Reopened

```
Offenses:
app/helpers/users_helper.rb:99:15: C: Style/FormatString: Favor format over sprintf.
      sprintf('%.2f', v).gsub('.', l(:general_csv_decimal_separator))
      ^^^^^^^
889 files inspected, 1 offense detected
```

#### #7 - 2020-04-04 15:06 - Go MAEDA

The following code fixes the Rubocop offence:

```
Index: app/helpers/users_helper.rb
=====
--- app/helpers/users_helper.rb      (000000 19642)
+++ app/helpers/users_helper.rb      (000000)
@@ -96,7 +96,7 @@
         format_object(value, false) do |v|
           case v.class.name
             when 'Float'
-             sprintf('%.2f', v).gsub('.', l(:general_csv_decimal_separator))
+             format('%.2f', v).gsub('.', l(:general_csv_decimal_separator))
           else
             v
           end
```

#### #8 - 2020-04-04 16:57 - Marius BĂLTEANU

Go MAEDA wrote:

The following code fixes the Rubocop offence:

[...]

Yes, this is an option, but format is just an alias of sprintf and it's not used anywhere else in the code. I propose to stick to sprintf and fix this using the following code:

```
mariusbalteanu@Mariuss-MacBook-Pro redmine % git diff
diff --git a/.rubocop.yml b/.rubocop.yml
index 73f1d18f0..4b63e0b5f 100644
--- a/.rubocop.yml
+++ b/.rubocop.yml
@@ -83,6 +83,9 @@
@@ -83,6 +83,9 @@ Style/AsciiComments:
Style/EmptyElse:
  EnforcedStyle: empty

+Style/FormatString:
+  EnforcedStyle: sprintf
+
+Style/FormatStringToken:
+  Enabled: false
```

In another ticket, we should replace in the code the "percent" style used in some files with sprintf and remove the todo from rubocop.

#### #9 - 2020-04-04 17:03 - Marius BĂLTEANU

- File 33126.patch added

#### #10 - 2020-04-05 06:30 - Go MAEDA

- Status changed from Reopened to Closed

Marius BALTEANU wrote:

Go MAEDA wrote:

The following code fixes the Rubocop offence:

[...]

Yes, this is an option, but format is just an alias of sprintf and it's not used anywhere else in the code. I propose to stick to sprintf and fix this

using the following code:  
[...]

Done in [r19643](#). Thank you.

#### #11 - 2020-05-04 02:27 - Go MAEDA

- Status changed from Closed to Reopened

I think it is better to sort columns for custom fields according to the position value, as issues CSV do.

```
diff --git a/app/helpers/users_helper.rb b/app/helpers/users_helper.rb
index 2a72a5db4..264259d5b 100644
--- a/app/helpers/users_helper.rb
+++ b/app/helpers/users_helper.rb
@@ -82,7 +82,7 @@ module UsersHelper
     'last_login_on',
     'status'
   ]
-  user_custom_fields = UserCustomField.all
+  user_custom_fields = UserCustomField.sorted

   # csv header fields
   csv << columns.map {|column| l('field_' + column)} + user_custom_fields.pluck(:name)
```

#### #12 - 2020-05-09 04:53 - Go MAEDA

- Status changed from Reopened to Closed

Go MAEDA wrote:

I think it is better to sort columns for custom fields according to the position value, as issues CSV do.  
[...]

Committed the fix in [r19767](#).

#### Files

feature-33126.patch	2.53 KB	2020-03-11	Mizuki ISHIKAWA
33126.patch	954 Bytes	2020-04-04	Marius BĂLTEANU