Redmine - Feature #41976

Ruby 3.4 support

2024-12-13 03:45 - Go MAEDA

Status:	New	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	Ruby support	**Estimated time:**	0.00 hour
Target version:	6.1.0		
Resolution:			

Description

Ruby 3.4.0 will be released on December 25, 2024.

Related issues:

Copied from Redmine - Feature #39761: Ruby 3.3 support	**Closed**

History

#1 - 2024-12-13 03:45 - Go MAEDA

- Copied from Feature #39761: Ruby 3.3 support added

#2 - 2024-12-13 08:19 - Go MAEDA

- Resolution deleted (Fixed)

#3 - 2024-12-13 08:20 - Go MAEDA

- File 0001-Fix-Avoid-literal-string-will-be-frozen-in-the-futur.patch added

The attached patch fixes "literal string will be frozen in the future" warnings on Ruby 3.4.0 rc1.

```
/path/to/redmine/app/views/gantts/show.html.erb:342: warning: literal string will be frozen in the future (run
 with --debug-frozen-string-literal for more information)

/path/to/redmine/app/views/repositories/_breadcrumbs.html.erb:18: warning: literal string will be frozen in th
e future (run with --debug-frozen-string-literal for more information)

/path/to/redmine/lib/redmine/i18n.rb:154: warning: warning: string returned by :ar.to_s will be frozen in the
future
```

#4 - 2024-12-16 01:51 - Katsuya HIDAKA

+1

0001-Fix-Avoid-literal-string-will-be-frozen-in-the-futur.patch

```
-      clss << " nwday" if @gantt.non_working_week_days.include?(wday)
+      clss += " nwday" if @gantt.non_working_week_days.include?(wday)
```

How about modifying it like this?

```
       clss = "gantt_hdr"
       clss << " nwday" if @gantt.non_working_week_days.include?(wday)
     %>
-    <%= content_tag(:div, :style => style, :class => clss) do %>
+    <%= content_tag(:div, :style => style, :class => ["gantt_hdr", { nwday: @gantt.non_working_week_days.incl
ude?(wday) }]) do %>
       <%= day_num.day %>
     <% end %>
     <%
```

With this change, the class attribute would produce the following results:

```
redmine-app(dev)> helper.content_tag(:div, "", :class => ["gantt_hdr", { nwday: false }])
=> "<div class=\"gantt_hdr\"></div>"
redmine-app(dev)> helper.content_tag(:div, "", :class => ["gantt_hdr", { nwday: true }])
```

```
=> "<div class=\"gantt_hdr nwday\"></div>"
```

Alternatively, this approach would also work:

```
-       clss = "gantt_hdr"
+       clss = +"gantt_hdr"
        clss << " nwday" if @gantt.non_working_week_days.include?(wday)
```

Reasons:

- The current implementation in the patch always creates a new string instance.
- Other similar parts of the code use String#+@, so this would improve consistency.

Personally, I think the former approach is preferable, but the latter also seems like a valid option.

#5 - 2025-02-14 07:26 - Katsuya HIDAKA

- File Add-Ruby-3.4-to-supported-Ruby-versions.patch added

Redmine appears to work fine with Ruby 3.4.

- I confirmed that the patch in #note-3 works correctly.
- All tests, including system tests run in CI, passed.
  https://github.com/hidakatsuya/redmine/actions/runs/13323213831
- It seems that there are no warnings or errors in the test results that might cause issues (Ruby warnings are enabled).

I am attaching a patch to add Ruby 3.4 to the supported Ruby versions.

Files

0001-Fix-Avoid-literal-string-will-be-frozen-in-the-futur.patch	2.36 KB	2024-12-13	Go MAEDA
Add-Ruby-3.4-to-supported-Ruby-versions.patch	1.43 KB	2025-02-14	Katsuya HIDAKA