Decide hexadecimal-encoded literals in order to be frozen string literals friendly

Currently, hexadecimal-encoded literals are used in many places in the source code of Redmine. I propose to rewrite those literals to UTF-8.

The reason is that those hexadecimal-encoded strings increase the work required to support frozen-string-literal which will be default in Ruby 3.0. For example, the following code (test/unit/mail_handler_test.rb:714) is not frozen-string-literal ready because you cannot call String#force_encoding for frozen strings.

```ruby
def test_add_issue_with_japanese_subject
  issue = submit_email('subject_japanese_1.eml', :issue => {:project => 'ecookbook'})
  assert_kind_of Issue, issue
  ja = "%xe3%83%86%e3%82%b9%e3%83%88".force_encoding('UTF-8')
  assert_equal ja, issue.subject
end
```

You will see the following error If you run the test.

```
Error:
MailHandlerTest#test_add_issue_with_japanese_subject:
FrozenError: can't modify frozen String
  test/unit/mail_handler_test.rb:721:in `force_encoding'
  test/unit/mail_handler_test.rb:721:in `test_add_issue_with_japanese_subject'
```

So, we have to update the code in the near future, at the latest until Ruby 3.0 is released. The patch provided by Pavel Rosický in #26561#note-9 fixes the code as follows:

```erb
diff --git a/test/unit/mail_handler_test.rb b/test/unit/mail_handler_test.rb
index adc4b71cf..75ebe3a30 100644
--- a/test/unit/mail_handler_test.rb
+++ b/test/unit/mail_handler_test.rb
@@ -1,3 +1,4 @@
+
+  # frozen-string-literal: true
+  # encoding: utf-8
+
+  # Redmine - project management software
@@ -717,7 +718,7 @@ class MailHandlerTest < ActiveSupport::TestCase
  :issue => {:project => 'ecookbook'}
 )
  assert_kind_of Issue, issue
- ja = "%xe3%83%86%e3%82%b9%e3%83%88".force_encoding('UTF-8')
+ ja = +("%xe3%83%86%e3%82%b9%e3%83%88").force_encoding('UTF-8')
  assert_equal ja, issue.subject
end
```

The code does not have any problem but there is an easier solution. We can make the code frozen-string-literal ready by just stopping using hexadecimal-encoded strings like the following. Even better, "テスト" that means "test" in Japanese is easier to read than "%xe3%83%86%e3%82%b9%e3%83%88". Of course, I can understand "テスト" in a moement but cannot understand "%xe3%83%86...".

```erb
diff --git a/test/unit/mail_handler_test.rb b/test/unit/mail_handler_test.rb
```

```ruby
def test_add_issue_with_japanese_subject
  issue = submit_email('subject_japanese_1.eml', :issue => {:project => 'ecookbook'})
  assert_kind_of Issue, issue
  ja = "テスト".force_encoding('UTF-8')
  assert_equal ja, issue.subject
end
```
There is no disadvantage at all to change hexadecimal-encoded strings to UTF-8 strings. I think we should complete this before working on #26561 (frozen-string-literal).

Related issues:
Related to Redmine - Feature #26561: Enable frozen string literals

Associated revisions
Revision 17991 - 2019-03-19 16:43 - Go MAEDA
Decode hexadecimal-encoded literals in order to be frozen string literals friendly (#31004).

Patch by Yuichi HARADA.

History
#1 - 2019-03-10 14:45 - Go MAEDA
- Related to Feature #26561: Enable frozen string literals added

#2 - 2019-03-10 15:13 - Go MAEDA
- Tracker changed from Defect to Feature

#3 - 2019-03-11 01:04 - Pavel Rosicky
thanks for poiting this out. I would definitely prefer this solution.

#4 - 2019-03-11 01:37 - Go MAEDA
- Subject changed from Change hexadecimal-encoded strings to UTF-8 string to Decode hexadecimal-encoded literals in order to be frozen string literals friendly

#5 - 2019-03-16 10:24 - Go MAEDA
- Assignee set to Go MAEDA

#6 - 2019-03-19 03:36 - Yuichi HARADA
- File 31004-hexadecimal-encoded-to-frozen-string.patch added

Most of the hexadecimal-encoded literals are under the test directory. Replaced UTF-8's hexadecimal-encoded literals in the test directory with string literals. Hexadecimal-encoded literals other than UTF-8 are mutable using String#+@ (https://docs.ruby-lang.org/en/2.6.0/String.html#method-i-2B-40). I attached a patch.

#7 - 2019-03-19 16:43 - Go MAEDA
- Status changed from New to Closed
- Target version set to 4.1.0
- Resolution set to Fixed
Committed the patch. Thanks.

#8 - 2019-03-20 06:44 - Go MAEDA
- Tracker changed from Feature to Patch

Files

| 31004-hexadecimal-encoded-to-frozen-string.patch | 63.5 KB | 2019-03-19 | Yuichi HARADA |

2024-04-13