

Redmine - Patch #34750

Remove unsupported encodings ISO-2022-KR and ISCII91 from Setting::ENCODINGS

2021-02-15 17:11 - Go MAEDA

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Go MAEDA	% Done:	0%
Category:	Code cleanup/refactoring	Estimated time:	0.00 hour
Target version:	4.2.0		

Description

ISO-2022-KR and ISCII91 are no longer supported by Ruby. These encodings should be removed from Setting::ENCODINGS constant.

Iconv library used in Redmine 2.6 and earlier supported these two encodings, but String#encode used in Redmine 3.0 and later does not.

```
$ ruby -e '"foo".encode("ISO-2022-KR")'
Traceback (most recent call last):
  1: from -e:1:in `'
-e:1:in `encode': code converter not found (UTF-8 to ISO-2022-KR) (Encoding::ConverterNotFoundError)

$ ruby -e '"foo".encode("ISCII91")'
Traceback (most recent call last):
  1: from -e:1:in `'
-e:1:in `encode': code converter not found (UTF-8 to ISCII91) (Encoding::ConverterNotFoundError)

$ ruby -e 'p Encoding.list.map(&:to_s)'
["ASCII-8BIT", "UTF-8", "US-ASCII", "UTF-16BE", "UTF-16LE", "UTF-32BE", "UTF-32LE", "UTF-16", "UTF-32", "UTF8-MAC", "EUC-JP", "Windows-31J", "Big5", "Big5-HKSCS", "Big5-UAO", "CP949", "Emacs-Mule", "EUC-KR", "EUC-TW", "GB2312", "GB18030", "GBK", "ISO-8859-1", "ISO-8859-2", "ISO-8859-3", "ISO-8859-4", "ISO-8859-5", "ISO-8859-6", "ISO-8859-7", "ISO-8859-8", "ISO-8859-9", "ISO-8859-10", "ISO-8859-11", "ISO-8859-13", "ISO-8859-14", "ISO-8859-15", "ISO-8859-16", "KOI8-R", "KOI8-U", "Shift_JIS", "Windows-1250", "Windows-1251", "Windows-1252", "Windows-1253", "Windows-1254", "Windows-1257", "IBM437", "IBM737", "IBM775", "CP850", "IBM852", "CP852", "IBM855", "CP855", "IBM857", "IBM860", "IBM861", "IBM862", "IBM863", "IBM864", "IBM865", "IBM866", "IBM869", "Windows-1258", "GB1988", "macCentEuro", "macCroatian", "macCyrillic", "macGreek", "macIceland", "macRoman", "macRomania", "macThai", "macTurkish", "macUkrainian", "CP950", "CP951", "IBM037", "stateless-ISO-2022-JP", "eucJP-ms", "CP51932", "EUC-JIS-2004", "GB12345", "ISO-2022-JP", "ISO-2022-JP-2", "CP50220", "CP50221", "Windows-1256", "Windows-1255", "TIS-620", "Windows-874", "MacJapanese", "UTF-7", "UTF8-DoCoMo", "SJIS-DoCoMo", "UTF8-KDDI", "SJIS-KDDI", "ISO-2022-JP-KDDI", "stateless-ISO-2022-JP-KDDI", "UTF8-SoftBank", "SJIS-SoftBank"]
$ ruby -e 'p Encoding.list.map(&:to_s)' | grep -ci -e ISO-2022-KR -e ISCII9
0

$ curl -O https://cache.ruby-lang.org/pub/ruby/2.7/ruby-2.7.2.tar.gz
$ tar xvf ruby-2.7.2.tar.gz
$ cd ruby-2.7.2
$ grep -ir -e ISO-2022-KR -e ISCII9 *
$
```

Associated revisions

Revision 20747 - 2021-02-17 13:24 - Go MAEDA

Remove unsupported encodings ISO-2022-KR and ISCII91 from Setting::ENCODINGS (#34750).

Patch by Go MAEDA.

History

#1 - 2021-02-15 17:17 - Go MAEDA

- Description updated

#2 - 2021-02-16 01:25 - Go MAEDA

Go MAEDA wrote:

Iconv library used in Redmine 2.6 and earlier supported these two encodings, but String#encode used in Redmine 3.0 and later does not.

This is not correct. Redmine from 2.3 to 2.6 uses Iconv only when it runs on Ruby 1.8 (see [source:tags/2.6.10/lib/redmine/codeset_util.rb#L1](https://source.ruby-lang.org/ruby-devel/2010/02/10/lib/redmine/codeset_util.rb#L1)).

Redmine version	Ruby version	uses Iconv?
< 2.3	-	yes
2.3 to 2.6	1.8	yes
2.3 to 2.6	>= 1.9	no
>= 3.0	-	no

So, ISO-2022-KR and ISCII91 are available only when all of the following conditions are met.

- Redmine version is 2.6.x or earlier
- Ruby version is 1.8.x or earlier

#3 - 2021-02-16 10:51 - Go MAEDA

- Subject changed from *Remove unsupported ISO-2022-KR and ISCII91 from Setting::ENCODINGS* to *Remove unsupported encodings ISO-2022-KR and ISCII91 from Setting::ENCODINGS*

- Target version set to 4.2.0

Setting the target version to 4.2.0.

#4 - 2021-02-17 13:24 - Go MAEDA

- Status changed from *New* to *Closed*

- Assignee set to *Go MAEDA*

Committed the patch.

Files

remove-unsupported-encodings.patch	652 Bytes	2021-02-15	Go MAEDA
------------------------------------	-----------	------------	----------