Ruby - Bug #4167

URI.encode encodes reserved character of

12/18/2010 12:07 AM - harking (George M. Harkin)

Status:	Closed		
Priority:	Normal		
Assignee:			
Target version:			
ruby -v:	ruby 1.9.2p0 (2010-08-18 revision 29036) [x86_64-linux]	**Backport:**	

Description

=begin
URI.encode's default behavior is to follow RFC 2732 [http://tools.ietf.org/html/rfc2732] which includes # in the list of characters to URI encode.

The updated RFC 3896 [http://tools.ietf.org/html/rfc3986#section-2.2] includes # in the list of reserved characters.

This bug is present in Ruby 1.8 too.

Observed behavior:

require 'uri'
enc = URI.encode("http://google.com/moo?testo=true#anchor21")
=> "http://google.com/moo?testo=true%23anchor21"

Expected behavior:

require 'uri'
enc = URI.encode("http://google.com/moo?testo=true#anchor21")
=> "http://google.com/moo?testo=true#anchor21"
=end

History

#1 - 12/18/2010 12:43 AM - harking (George M. Harkin)

=begin
Workaround is to use

exp = URI.encode(x, Regexp.new("[^#{URI::PATTERN::UNRESERVED}#{URI::PATTERN::RESERVED}#]", false, 'N'))

where x is the URL with a # symbol for an anchor tag

=end

#2 - 12/18/2010 01:12 AM - harking (George M. Harkin)

=begin
Just noticed "warn "#{caller(1)[0]}: warning: URI.escape is obsolete" if $VERBOSE".

What is the replacement at this time?

=end

#3 - 12/18/2010 06:19 PM - naruse (Yui NARUSE)

=begin
URI lib says it refers RFC2396, so current behavior is correct in its spec.

Yes, I know current behavior is not what you expect.
So we plan to change the lib to refer RFC3986.

Moreover current URI.encode is simple gsub.
But I think it should split a URI to components, then escape each components,
and finally join them.

So current URI.encode is considered harmful and deprecated.
This will be removed or change behavior drastically.

> What is the replacement at this time?

As I said above, current URI.encode is wrong on spec level.
So we won't provide the exact replacement.
The replacement will vary by its use case.

We thought most use case is to generate escaped URI from joined URI componets.
For this, people should use URI.join or URI.encode_www_form;
you should escape each components before join them.
=end

#4 - 06/26/2011 04:33 PM - naruse (Yui NARUSE)

- Status changed from Open to Closed

URI.encode is obsoleted.

Note that HTML5 seems specify the behavior of escaping and unescaping for addressbar.
Future URI module may have such methods.