Ruby master - Feature #4513
allow whitespace following EOL continuation backslash
03/21/2011 09:14 AM - Rich_Morin (Rich Morin)

Status: Closed
Priority: Normal
Assignee: nobu (Nobuyoshi Nakada)
Target version:

Description
In a few programming languages, the displayed or printed program can hide
critical syntactic and/or semantic information. For example, in make(1)
and Python, the difference between initial tabs and spaces can cause bugs
that are not visually apparent.

Ruby has a minor instance of this problem, in that spaces or tabs
between a backslash and the EOL will prevent the line from being continued:

```ruby
>> a \
?> = 2
=> 2
```

```ruby
>> b \
SyntaxError: compile error
(irb):3: syntax error, unexpected $undefined, expecting $end
from (irb):3
```

Could Ruby open up its syntax to allow any number of intervening spaces or tabs before the EOL?

Related issues:
Related to Ruby master - Bug #14597: Ripper Issue - introduced between 62674 ...

Associated revisions
Revision 62178 - 02/03/2018 07:30 AM - nobu (Nobuyoshi Nakada)
parse.y: named backslash
- parse.y: named backslash to show unexpected character.

Revision 62663 - 03/05/2018 08:32 AM - nobu (Nobuyoshi Nakada)
parse.y: named whitespaces
- parse.y: named escaped whitespaces to show unexpected character. bare whitespaces should not appear outside of word_list.

Revision 62675 - 03/06/2018 09:03 AM - nobu (Nobuyoshi Nakada)
parse.y: tSPACE for old bison
- parse.y (tSPACE): define a separate token for escaped space, to fix redefining user token number of " " error at word list separator on bison 2.

History
#1 - 03/18/2012 07:12 PM - nahi (Hiroshi Nakamura)
- Description updated
- Category set to core
- Status changed from Open to Assigned
- Assignee set to nobu (Nobuyoshi Nakada)

Regardless it's acceptable or not, we want to check nobu's local patches first. Don't you have that?

#2 - 11/20/2012 09:28 PM - mame (Yusuke Endoh)
- Target version set to 2.6
It is easy to change, but I doubt if it is useful.

Warning for a backslash followed by spaces is:

Old:

```
$ ruby2.5 -e ' \n'
-e:1: syntax error, unexpected $undefined
```

Now:

```
$ ruby -e ' \n'
-e:1: syntax error, unexpected backslash
```

Does this help you?

Close this for now.

Related to Bug #14597: Ripper Issue - introduced between 62674 & 62681 added

Now:

```
$ ruby -e ' \n'
-e:1: syntax error, unexpected escaped space
```

Close this for now.

Description updated