Sed Command in Linux - Append and Insert Lines to a File

Unknown April 19, 2015 linux commands, sed 18 comments

This is the second article of the **"Super sed' Series"**, in which we will learn how to append and insert lines to a file using line numbers and regular expressions. In the previous article in the series, we learned to print lines in a file using sed command.

Before we directly jump to the main content, every learner should know what sed is. Here is the brief introduction of the Super sed:

- sed stand for **S**tream **ED**itor and it being based on the ed editor, it borrows most of the commands from the ed. It was developed by Lee E. McMahon of Bell Labs.

- `sed` offers large range of text transformations that include printing lines, deleting lines, editing line in-place, search and replace, appending and inserting lines, etc.
- `sed` is useful whenever you need to perform common editing operations on multiple lines without using 'vi' editor.
- Whenever `sed` is executed on an input file or on the contents from stdin, `sed` reads the file line-by-line and after removing the trailing newline, places it in the "Pattern space", where the commands are executed on them after conditions (as in case of regex matching) are verified, and then printed on the stdout.

Before we start, just remember two points:

1. `sed` **"a"** command lets us append lines to a file, based on the line number or regex provided. So, the lines will be added to the file AFTER the line where condition matches.
2. `sed` **"i"** command lets us insert lines in a file, based on the line number or regex provided. So, the lines will be added to the file AT the location where line number matches or BEFORE the line where pattern matches.
3. `sed` with option `-i` will edit the file in place, i.e. unless you use the option `-i`, the changes will not be written to the file. (Explained in later section)

`sed` - Appending Lines to a File

For our better understanding, let us have a file `sedtest.txt` with contents as follows:

```
$ cat sedtest.txt
This is line #1
This is line #2
This is line #3
This is line #4
This is line #5
This is line #6
This is line #7
This is line #8
```

```
This is line #9
This is line #10
```

1. Append a line after 'N'th line

This will add a line after 'N'th line in the `FILE.txt`.

Syntax:

```
sed 'N a <LINE-TO-BE-ADDED>' FILE.txt
```

Example:

To append a line **`#This is just a commented line`** after 1st line,

```
$ sed '1 a #This is just a commented line' sedtes
This is line #1
#This is just a commented line
This is line #2
This is line #3
This is line #4
This is line #5
This is line #6
This is line #7
This is line #8
This is line #9
This is line #10
```

While, to append a line after last line,

```
$ sed '$ a This is the last line' sedtest.txt
This is line #1
This is line #2
This is line #3
This is line #4
This is line #5
This is line #6
This is line #7
This is line #8
This is line #9
This is line #10
This is the last line
```

If you run above commands and inspect the file `sedtest.txt`, you would find that, the original contents of that file would not change. In case you wish to append lines in the file and save the changes (i.e. edit the file in place), you will have to use the option `-i`.

Lets check it for the latest command we have run to append lines after the last line of the file. Has it made any changes to the file?

```
$ cat sedtest.txt
This is line #1
This is line #2
This is line #3
This is line #4
This is line #5
This is line #6
This is line #7
This is line #8
This is line #9
This is line #10
```

No, the original file remains the same. But, I wanted to save the changes to the file. So, I should have used the option `-i`.

```
$ sed -i '$ a This is the last line' sedtest.txt
$ cat sedtest.txt
This is line #1
This is line #2
This is line #3
This is line #4
This is line #5
This is line #6
This is line #7
This is line #8
This is line #9
This is line #10
This is the last line
```

Yes, now changes are written to the file. Just remember this.

2. Append Line using Regular Expression/Pattern

This will append the line after the line where pattern match is found.

Syntax:

```
sed '/PATTERN/ a <LINE-TO-BE-ADDED>' FILE.txt
```

Example:

```
$ sed '/5/ a #Next line is the 6th line, not this
This is line #1
This is line #2
This is line #3
This is line #4
This is line #5
#Next line is the 6th line, not this
This is line #6
This is line #7
This is line #8
This is line #9
This is line #10
```

`sed` - Inserting Lines in a File

1. Insert line using the Line number

This will insert the line before the line at line number 'N'.

Syntax:

```
sed 'N i <LINE-TO-BE-ADDED>' FILE.txt
```

Example:

```
$ sed '4 i #This is the extra line' sedtest.txt
This is line #1
This is line #2
This is line #3
#This is the extra line
This is line #4
```

```
This is line #5
This is line #6
This is line #7
This is line #8
This is line #9
This is line #10
```

While, to insert a line before last line,

```
$ sed '$ i #Next line will be last line' sedtest.
This is line #1
This is line #2
This is line #3
This is line #4
This is line #5
This is line #6
This is line #7
This is line #8
This is line #9
#Next line will be last line
This is line #10
```

2. Insert lines using Regular expression

This will insert the line before every line where pattern match is found.

Syntax:

```
sed '/PATTERN/ i <LINE-TO-BE-ADDED>' FILE.txt
```

Example:

```
$ sed '/8/ i #This line is inserted using sed' se
This is line #1
This is line #2
This is line #3
This is line #5
This is line #6
This is line #7
#This line is inserted using sed
This is line #8
```

```
This is line #9
This is line #10
```

That's all about the second article on sed command. More articles on sed are coming soon. So, stay tuned. Of course, do not forget to share your feedback in the comment section below.

Related Posts:

Sed Command in Linux - Search and Replace Text in a File

This is the fourth article of the "Super sed' Series", in which we will learn how to search for a pattern and replace it with other pattern in a file,… Read More

Sed Command in Linux - Delete Lines from a File

This is the third article of the "Super sed' Series", in which we will learn how to delete lines from a file using line numbers and regular expr… Read More

Sed Command in Linux - Append and Insert Lines to a File

This is the second article of the "Super sed' Series", in which we will learn how to append and insert lines to a file using line numbers and regular … Read More

Sed Command in Linux - Print Lines in a File

This is the first article of the "Super sed' Series", in which we will learn how to print lines in a file using line numbers and regular expressions. … Read More

18 comments:

Anonymous 4 February 2016 at 19:38

Thank you

Reply

Unknown 12 February 2016 at 12:12

That was very useful..thanks

Reply

wolfie 15 February 2016 at 14:15

How can I replace a text with the line number by using only sed?
I have a .sed config file that generates an csv from a log file. I want to add a column that is equal to a formula that uses the previous column as argument. So for line 4 =B4, for line 5 =B5 etc

Reply

Anonymous 16 February 2016 at 20:31

Thanks for the useful post :)

Reply

Anonymous 22 May 2016 at 13:52

Thanks for the useful post. Congratulations

Reply

Unknown 28 July 2016 at 19:51

Big Thanks :)

Reply

Anonymous 15 September 2016 at 18:33

Good Info. Thank you.

Reply

Anonymous 16 September 2016 at 19:23

ls | sed '2 i ExtraLine'
sed: command garbled: 2 i ExtraLine

Reply

SRINIVAS DARIPELLI 27 January 2017 at 22:38

Interesting points mentioned on sed usage.one of the best articles here mentioned.love to see more of this kind going forward.

Reply

Unknown 14 March 2017 at 19:05

I have file A with 100 lines and file B with 10 lines. I want to insert all the lines from file B in to file A starting 80th line.

Reply

Unknown 6 July 2017 at 16:22

I would like to insert sed '$ a test_user: 'test'' a line with quotes. but it is getting ignored when i ran the above command and it inserted without quotes. Please suggest.

Reply

Tanaji Kamble 11 August 2017 at 16:02

i want to count the total no of lines in file and add that count value to first line. The script written is as given below

!/bin/bash
#delete the first 11 lines and keep back up of original file $1

sed -i.bak '1,11 d' $1

#count the total lines and assigned to variable k

k= sed -n '$=' $1

#print k

echo $k

#insert value of k at line no 1

sed -i "1i '$k'" $1

the last command is unable to insert value of variable k at line 1 in file specified by argument $1

Reply

Anonymous 14 September 2017 at 01:09

Very helpful in my task to add line after PATTERN

Reply

Roman Kahramanı değilim... 8 December 2017 at 21:17

Thanks is useful things. how to insert blank line for example; between first line and second line?

Reply

anon 23 January 2018 at 16:17

Good information. Thank you but I need sed '$ i #Next line will be last line "current time"' sedtest.txt

Reply

Anonymous 17 February 2018 at 12:38

It is good,but not so good,it is kinda bad,but not that bad,it is useful,but not that useful.

Great! :)

Reply

Bisbick 12 April 2018 at 21:36

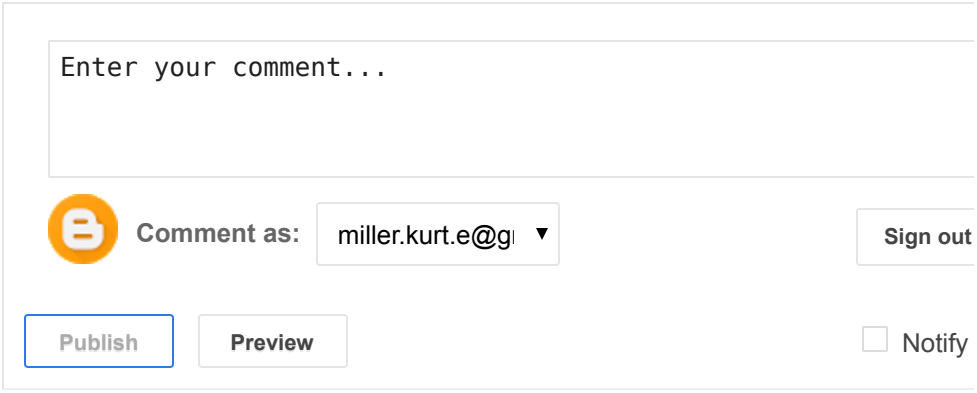
Thasnks, very useful and simplify my work

Reply

Servie to the Nation 4 August 2018 at 16:09

thanks You

Reply



Links to this post

Create a Link

Subscribe to: Post Comments (Atom)