## How to append output to the end of a text file

Ask Question



How do I append the output of a command to the end of a text file?

349

bash shell



44

edited Oct 23 '18 at 23:59



asked Jun 1 '11 at 20:55

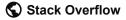


**1,784** 3 12 5

## 8 Answers

Home

**PUBLIC** 



Tags

Users

Jobs



Use >> instead of > when directing output to a file:

your\_command >> file\_to\_append\_to

If file\_to\_append\_to does not exist, it will be created.

## **Example:**

```
echo "hello" > file
$ echo "world" >> file
$ cat file
```

edited Jul 14 '16 at 12:29

answered Jun 1 '11 at 20:57



aioobe

**333k** 77 704 759

- The problem is that echo removes the newlines from the string. How do you append to a file a string which contains newlines? Timothy Swan Dec 15 '17 at 21:25
- 5 @TimothySwan, I believe with the -e option. Valentin Grégoire Mar 5 '18 at 7:44



You can use the >> operator. This will append data from a command to the end of a text file.

66

To test this try running:



echo "Hi this is a test" >> textfile.txt

Do this a couple of times and then run:

cat textfile.txt

You'll see your text has been appended several times to the textfile.txt file.

answered Jun 1 '11 at 20:58



wal

**,152** 16 2



To append a file use >>

52

```
echo "hello world" >> read.txt
cat read.txt
echo "hello siva" >> read.txt
cat read.txt
```

then the output should be

```
hello world # from 1st echo command
hello world # from 2nd echo command
hello siva
```

To overwrite a file use >

```
echo "hello tom" > read.txt
cat read.txt
```

then the out put is

hello tom

edited Jan 26 at 5:54

Community 

1 1

answered Apr 30 '18 at 10:16





Use command >> file\_to\_append\_to to append to a file.



**CAUTION:** if you only use a single > you will completely overwrite the contents of the file. To ensure that doesn't ever happen, you can add set -o noclobber to your .bashrc .

This ensures that if you accidentally type <code>command > file\_to\_append\_to</code> to an existing file, it will alert you that the file exists already. Sample error message: <code>file exists: testFile.txt</code>

Thus, when you use > it will only allow you to create a new file, not overwrite an existing file.

answered Oct 21 '17 at 13:59



raft

**43** 6



Use the >> operator to append text to a file.

20



answered Jun 1 '11 at 20:57



Nev Stokes

**5,710** 3 31 39



for the whole question:

13

cmd >> o.txt && [[ \$(wc -1 < o.txt) -eq 720 ]] && mv o.txt '</pre>



this will append 720 lines (30\*24) into o.txt and after will rename the file based on the current date.

Run the above with the cron every hour, or

while .

sleep 3600 done

answered Jun 1 '11 at 22:56





I'd suggest you do two things:

7

- 1. Use >> in your shell script to append contents to particular file. The filename can be fixed or using some pattern.
- 2. Setup a hourly cronjob to trigger the shell script

answered Sep 11 '14 at 15:20

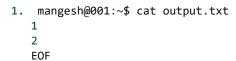


sballmer



For example your file contains:







if u want to append at end of file then ---->remember spaces between 'text' >> 'filename'

2. mangesh@001:~\$ echo somthing to append >> output.txt|
1
2
EOF

## And to overwrite contents of file:

3. mangesh@001:~\$ echo 'somthing new to write' > output
somthing new to write

edited Jan 20 '17 at 15:50



Torsten

**737** 1 11 35

answered Jan 15 '15 at 7:38



Mangesh Bhapkar

This is misleading in many details. Spaces are *not* important and piping an empty output to <code>cat</code> is ... just completely wacky. (It's empty because you just redirected standard output to a file.) – tripleee Oct 24 '18 at 4:24