

Linux and Unix `fc` command tutorial with examples

Tutorial on using `fc`, a UNIX and Linux command for editing and re-executing commands previously entered into an interactive shell. Examples of editing and re-executing the last command, editing and executing a previous command, setting the text editor to be used, listing previous commands and executing a command without editing it.

Estimated reading time: 4 minutes

Table of contents

- [What is the `fc` command?](#)
- [How to edit and re-executing the last command](#)
- [How to list previous commands](#)
- [How to edit and re-execute a previous command](#)
- [How to set the text editor dynamically](#)
- [How to view when commands were run](#)
- [How to re-execute a command without editing it](#)
- [Further reading](#)

```
FC(1P)                POSIX Programmer's Manual                FC(1P)

PROLOG

    This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this
    interface may differ (consult the corresponding Linux manual page for details of Linux behavior),
    or the interface may not be implemented on Linux.

NAME

    fc – process the command history list
```

What is the `fc` command?

The `fc` command is a command line utility for listing, editing and re-executing commands previously entered into an interactive shell. The `fc` command is a [shell builtin](https://en.wikipedia.org/wiki/Fc_(Unix)) ([https://en.wikipedia.org/wiki/Fc_\(Unix\)](https://en.wikipedia.org/wiki/Fc_(Unix))) meaning the command comes from the shell rather than the operating system. As such `fc` can vary slightly depending on the shell being used. The `fc` command is present in most shells including `bash`, `zsh` and `ksh`. The `fc` command can be very useful for editing long commands entered into a shell and re-executing them without having to rewrite the entire command. The examples here are from the `zsh` shell.

How to edit and re-executing the last command

To edit and run the last command entered into a shell run the `fc` command. This will open a text editor and allow the command to be edited. When the file is saved the command will be run. This

can be very useful for editing and re-executing complicated commands in a shell.

```
echo 'last command'
last command
fc
...file edited, string changed to 'hello'..
echo 'hello'
hello
```

The text editor invoked by `fc` can be specified firstly in the `FCEDIT` environment variable. In the following example the `FCEDIT` environment variable is set to `nano`. When the `fc` command is run the `nano` text editor will be invoked. If `FCEDIT` is not set the `EDITOR` environment variable will be used. After that the `vi` command will be used.

```
export FCEDIT=nano
ls
fc
```

How to list previous commands

To list previous commands and not execute them pass the `-l` option. This will show a list of commands run in the current interactive shell.

```
fc -l
700 ip link set dev wlp3s0 down
701 sudo ip link set dev wlp3s0 down
702 ip a
703 dmesg
...
```

To start from a specific command pass the number to `fc` along with the `-l` option.

```
fc -l 600
600 cat /etc/modules-load.d/bcwc-pcie.conf
601 sudo reboot
602 startx
...
```

To list a specific range of commands the first and last numbers may also be passed.

```
fc -l 701 702
701 sudo ip link set dev wlp3s0 down
702 ip a
```

To suppress numbers in listings the `-n` can be used.

```
fc -ln 701 702
sudo ip link set dev wlp3s0 down
ip a
```

How to edit and re-execute a previous command

To edit and re-execute a previous command first find the command to edit using `fc -l`. Then pass the number to be edited and re-executed.

```
fc -l
841 echo 'hello'
842 echo 'world'
fc 841
echo 'hello'
hello
```

How to set the text editor dynamically

To set the text editor that is used to edit commands dynamically pass the editor to be used to the `-e` option. In the following example the `pwd` command will be edited in `vi`.

```
pwd
fc -e vi
```

On saving the file the command will be run.

How to view when commands were run

For the `zsh` shell it is possible to see when commands were run by passing the `-d` option to show the timestamp for each command. In the following example the `-l` options is also passed to only list commands.

```
fc -ld
700 14:43 ip link set dev wlp3s0 down
701 14:43 sudo ip link set dev wlp3s0 down
702 14:43 ip a
```

A full timestamp may be shown with the `-f` option.

```
fc -lf
700 9/28/2016 14:43 ip link set dev wlp3s0 down
701 9/28/2016 14:43 sudo ip link set dev wlp3s0 down
702 9/28/2016 14:43 ip a
```

For a European date format the `-E` option may be used.

```
fc -lE
fc -lE 700 702
700 28.9.2016 14:43 ip link set dev wlp3s0 down
701 28.9.2016 14:43 sudo ip link set dev wlp3s0 down
702 28.9.2016 14:43 ip a
```

How to re-execute a command without editing it

To re-execute a command without invoking an editor pass the `-e` option with a dash. This will cause the command to be run immediately.

```
echo 'hello'
hello
fc -e -
echo 'hello'
hello
```

This can also accept a command number or a range of numbers from the history.

```
fc -l
842 echo 'hello world'
843 fc -e -
844 echo 'hello'
fc -e - 842
echo 'hello world'
hello
```

Further reading

- [fc man page \(https://linux.die.net/man/1/fc\)](https://linux.die.net/man/1/fc)
- [fc Wikipedia page \(https://en.wikipedia.org/wiki/Fc_\(Unix\)\)](https://en.wikipedia.org/wiki/Fc_(Unix))
- [Linux and Unix fc and history commands \(http://www.computerhope.com/unix/uhistory.htm\)](http://www.computerhope.com/unix/uhistory.htm)

Have an update or suggestion for this article? [You can edit it here and send me a pull request.](https://github.com/shapeshed/shapeshed.com/edit/master/content/posts/unix-fc.md)
(<https://github.com/shapeshed/shapeshed.com/edit/master/content/posts/unix-fc.md>)

Tags

- [UNIX \(/tags/unix\)](/tags/unix)
- [Linux \(/tags/linux\)](/tags/linux)

Recent Posts

About the author

George Ornbo is a UK based human.

He is interested in people, music, food and writing. In a previous version of himself he wrote [books \(https://www.amazon.com/Sams-Teach-Yourself-Hours-Programming/dp/0672338033\)](https://www.amazon.com/Sams-Teach-Yourself-Hours-Programming/dp/0672338033) on [technology \(https://www.amazon.com/Sams-Teach-Yourself-Node-js-Hours/dp/0672335956\)](https://www.amazon.com/Sams-Teach-Yourself-Node-js-Hours/dp/0672335956).

[← http://shapeshed.com \(/\)](http://shapeshed.com/)

Content is licensed under a Creative Commons [Attribution-NonCommercial-ShareAlike 4.0 International \(CC BY-NC-SA 4.0\)](https://creativecommons.org/licenses/by-nc-sa/4.0/) (<https://creativecommons.org/licenses/by-nc-sa/4.0/>)