

Ask Ubuntu is a question and answer site for Ubuntu users and developers. Join them; it only takes a minute:

Here's how it works:

Sign up

Anybody can ask a question

Anybody can answer


The best answers are voted up and rise to the top

How can I recursively delete all files of a specific extension in the current directory?

How do I safely delete all files with a specific extension (e.g. .bak) from current directory and all subfolders using one command-line? Simply, I'm afraid to use `rm` since I used it wrong once and now I need advice.

command-linefilesrmbatch

edited Nov 15 '13 at 16:16

 **Glutanimate**
14.4k 7 65 116

asked Nov 15 '13 at 13:03

 **user216038**
1,488 3 6 4

4 Answers

You don't even need to use `rm` in this case if you are afraid. Use `find` :

```
find . -name "*.bak" -type f -delete
```

But use it with precaution. Run first:

```
find . -name "*.bak" -type f
```

to see exactly which files you will remove.

Also, **make sure that `-delete` is the last argument** in your command. If you put it before the `-name *.bak` argument , **it will delete everything**.

See [man find](#) and [man rm](#) for more info and see also this related question on SE:

- How do I remove all .pyc files from a project?

edited May 23 at 12:39

 **Community** ♦
1

answered Nov 15 '13 at 13:08

 **Radu Rădeanu**
92.4k 27 206 288

	How's this different from <code>rm *.bak</code> ? – sayantankhan Nov 15 '13 at 13:11
6	@Bolt64 Your <code>rm *.bak</code> will not work for subdirectories. – Radu Rădeanu Nov 15 '13 at 13:14
	With default settings <code>rm *.bak</code> will only delete all files ending with <code>.bak</code> in the current directory. TO also do things in subdirectories you either needed to fiddle with globs, use the <code>-r</code> option or use the <code>find</code> example. – Hennes Nov 15 '13 at 13:14
7	@Hennes Be careful with <code>rm -r *.bak</code> ! It also removes directories ending in <code>.bak</code> with all their content. – Radu Rădeanu Nov 15 '13 at 13:34
15	Make sure that <code>-delete</code> is the last argument in your command. If you put it before the <code>-name *.bak</code> argument, <i>it will delete everything</i> . – Michael Oct 29 '14 at 14:36

```
find . -name "*.bak" -type f -print0 | xargs -0 /bin/rm -f
```

edited Feb 8 '15 at 16:13

 **muru**
97.2k 15 193 306

answered Apr 4 '14 at 19:10

 **lokera**
359 3 2

- 1 Welcome to askubuntu! While this is a perfectly valid answer I don't think there's any advantage in using this instead of the `-delete` flag of `find`. More information can be found in the [GNU manuals entry for deleting files with find](#). – [Glutanimate](#) Apr 4 '14 at 20:11
- 1 you are probably right, it's just an alternative solution, perhaps more raw ;) – [lokera](#) Apr 4 '14 at 22:13
- 7 This is not just an alternative but an example how other commands can be combined together with the pipe '|'. +1 – [Boris Pavlović](#) Jun 5 '14 at 7:18
- 8 This alternative solutions work on other environments that lack `-delete` (like cygwin) – [ciriarte](#) Aug 15 '14 at 4:07
- 1 I think this is the best answer here. – [Léo Léopold Hertz 준영](#) Jun 29 '15 at 11:27

First run the command `shopt -s globstar`. You can run that on the command line, and it'll have effect only in that shell window. You can put it in your `.bashrc`, and then all newly started shells will pick it up. The effect of that command is to make `**/` match files in the current directory and its subdirectories recursively (by default, `**/` means the same thing as `*/`: only in the immediate subdirectories). Then:

```
rm **/*.bak
```

(or `gvfs-trash **/*.bak` or what have you).

answered Nov 15 '13 at 18:59



Gilles

37.3k

9

90

129

+1 for just using rm command – Luca Steeb Sep 14 '16 at 16:32

Deleting files is for me not something you should use `rm` for. Here is an alternative:

```
find . -name "*.bak" -type f # display what will be deleted
sudo apt-get install gvfs    # install a tool that allows you to put stuff in the trash
alias "trash"="gvfs-trash"   # you can also put this in .bash_aliases or simply use the
command without alias
trash *.bak                  # trash the files (thus moving them to the trash bin)
```

As Flimm states in the comments:

The package `trash-cli` does the same thing as `gvfs-trash` without the dependency on `gvfs`.

edited Nov 20 '13 at 10:06

answered Nov 15 '13 at 13:52



don.joe

13.3k

8

51

89

2 "Don't use `rm` to delete things" is a controversial statement but I have to agree that it's often wiser to use something that will let you undo things. – Oli ♦ Nov 15 '13 at 15:12

1 The package `trash-cli` does the same thing as `gvfs-trash` without the dependency on `gvfs`. – Flimm Nov 20 '13 at 9:08

I have edited it in the answer, next time feel free to do the edit yourself. – don.joe Nov 20 '13 at 10:06