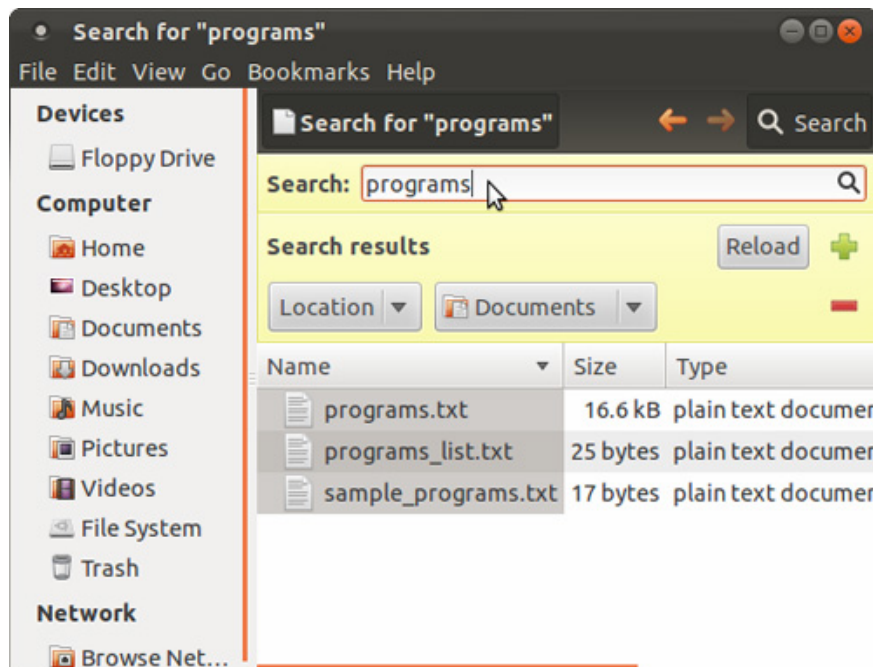


# Search for files and directories in Linux using the command line interface

You can use the graphical file management utility to find files in Linux like Nautilus in Gnome, Dolphin in KDE and Thunar in Xfce. However, there are several ways to use the command line to find files in any Linux desktop management utility.

You can use the graphical file management utility to find files in Linux like Nautilus in Gnome, Dolphin in KDE and Thunar in Xfce. However, there are several ways to use the command line to find files in any Linux desktop management utility.



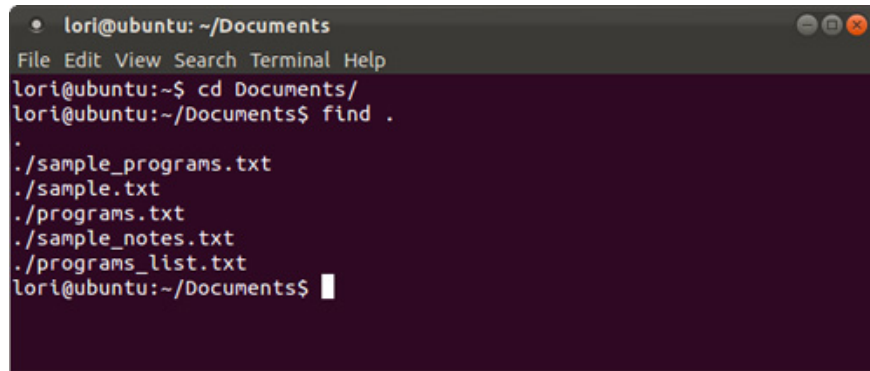
## Use the Find command

The Find command allows users to find files if they know the file name roughly. This is the simplest type of command that searches for files in the current directory that match certain criteria. Users can search for files by name, owner, group, file type, permissions, date and year and many other criteria.

Type the following command to list all files found in the current directory:

```
tim.
```

The dot indicates the current directory.

A terminal window titled 'lori@ubuntu: ~/Documents' with a menu bar (File, Edit, View, Search, Terminal, Help). The prompt is 'lori@ubuntu:~\$ cd Documents/'. The user enters 'lori@ubuntu:~/Documents\$ find .' and the output lists: '.', './sample\_programs.txt', './sample.txt', './programs.txt', './sample\_notes.txt', and './programs\_list.txt'. The prompt returns to 'lori@ubuntu:~/Documents\$'.

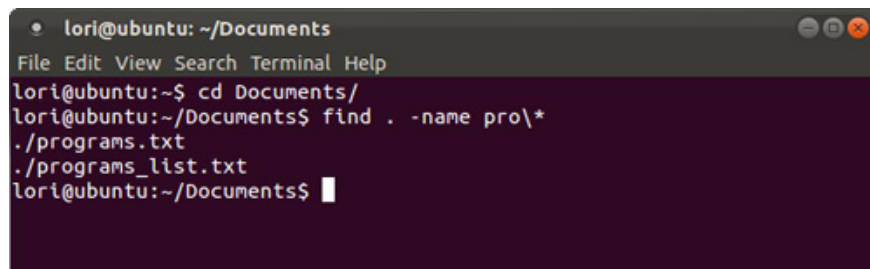
To find files that match a particular pattern, use the '*-name*' parameter. Users can use metacharacters (such as '\*' ) but must leave the escape symbol '\' before each super-character.

For example, if we want to find all the files that start with '*pro*' in the Documents folder, use **cd Documents /** to move to the Documents folder, then type the following command:

```
tim. -Name pro *
```

All files starting with '*pro*' in the directory will be listed.

Note: The default find statement is case sensitive. If you want to search for a word or phrase that is not case sensitive, use the '*-iname*' option with the find statement.

A terminal window titled 'lori@ubuntu: ~/Documents' with a menu bar (File, Edit, View, Search, Terminal, Help). The prompt is 'lori@ubuntu:~\$ cd Documents/'. The user enters 'lori@ubuntu:~/Documents\$ find . -name pro\\*' and the output lists: './programs.txt' and './programs\_list.txt'. The prompt returns to 'lori@ubuntu:~/Documents\$'.

If no file matches the criteria, no results will be displayed.

The find statement has many options for filtering searches. To find out more, type '*man find*' in the Terminal window and press **Enter** .

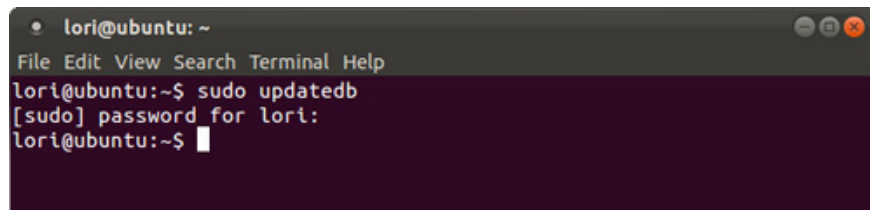
## Use the locate command

The locate command is faster than the find command because it uses the previously built database while the find command searches the system in real time, from all directories and files. The locate command returns a list of all path names that contain a group of special characters.

The database is updated periodically from *cron* , but users can update themselves at any time so they can get results immediately. To do this, type the following statement:

sudo updatedb

Enter password.

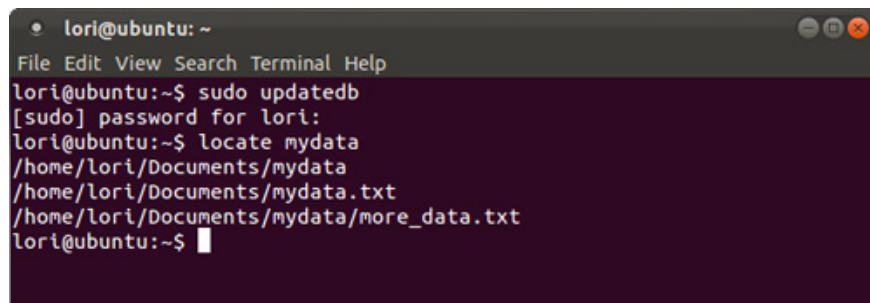


```
lori@ubuntu: ~  
File Edit View Search Terminal Help  
lori@ubuntu:~$ sudo updatedb  
[sudo] password for lori:  
lori@ubuntu:~$
```

The original form of the locate command finds all files in the file system, starting from the root directory that contains all or any part of the search criteria.

locate mydata

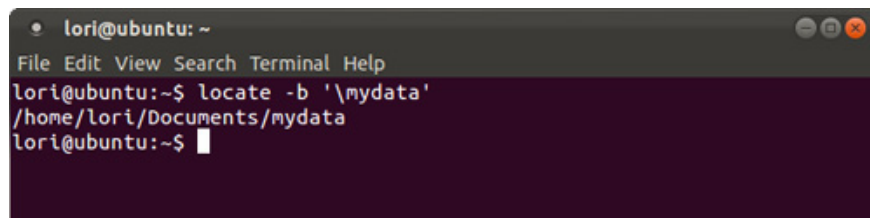
For example, the following statement found two files containing 'mydata' and a file containing 'data'.



```
lori@ubuntu: ~  
File Edit View Search Terminal Help  
lori@ubuntu:~$ sudo updatedb  
[sudo] password for lori:  
lori@ubuntu:~$ locate mydata  
/home/lori/Documents/mydata  
/home/lori/Documents/mydata.txt  
/home/lori/Documents/mydata/more_data.txt  
lori@ubuntu:~$
```

If you want to find all files or folders that contain exactly the search criteria, use the '-b' option with the locate command as follows:

locate -b'mydata '



```
lori@ubuntu: ~  
File Edit View Search Terminal Help  
lori@ubuntu:~$ locate -b '\mydata'  
/home/lori/Documents/mydata  
lori@ubuntu:~$
```

The command of mlocate is a variant of locate. Search results include only files that the current user has access to. When you update the mlocate database, it maintains the time stamp information in the database. This allows mlocate to know if the content in a directory is changed without re-reading the content and updating it to a faster database without relying on the hard drive.

When installing mlocate, the /usr/bin/locate binary file changes and points to mlocate. To install mlocate when the command is not in the Linux distribution, type the following command:

sudo apt-get install mlocate

```
lori@ubuntu: ~  
File Edit View Search Terminal Help  
lori@ubuntu:~$ sudo apt-get install mlocate  
[sudo] password for lori: █
```

The mlocate command does not use the same database file as the standard locate command. Therefore, users can create the database manually by typing the following statement:

```
sudo /etc/cron.daily/mlocate
```

The mlocate command will not work until the database is created manually or when the script is run from *cron* .

```
lori@ubuntu: ~  
File Edit View Search Terminal Help  
lori@ubuntu:~$ sudo /etc/cron.daily/mlocate  
lori@ubuntu:~$ █
```

For detailed information about locate and mlocate statements, type **man locate** or **man mlocate** in the command window and press **Enter** . The help screen will appear.

## Use the which command

The which command returns the absolute path of the executable file called when a command is issued. This is useful when you need to find the location of an executable file to create a shortcut to the program on the screen, on a panel or elsewhere in the desktop management utility. For example, typing the command **which firefox** will display the results as shown below.

Note: By default, the which command only displays the first match. To display the entire result, use the '*-a*' option with the command. The '*which firefox*' command will become '*which -a firefox*' .

```
lori@ubuntu: ~  
File Edit View Search Terminal Help  
lori@ubuntu:~$ which firefox  
/usr/bin/firefox  
lori@ubuntu:~$ █
```

Users can search multiple executable files at the same time. Only the paths to executable files will be displayed. In the example below, only the executable file '*ps*' is found.

```
lori@ubuntu: ~
File Edit View Search Terminal Help
lori@ubuntu:~$ which delete talk ps
/bin/ps
lori@ubuntu:~$
```

Note: The which command only finds the path to the executable file that the current user is allowed to use. If searching for an executable file that only the administrator is allowed to use, no results will be displayed.

To learn more about the which command, type **man which** at the command **prompt** and press **Enter** .

## Use the whereis command

The whereis command is used to find the location of binary files, source files, or 'man' files for a program. For example, when typing **whereis firefox** 'we will get the results shown below.

```
lori@ubuntu: ~
File Edit View Search Terminal Help
lori@ubuntu:~$ whereis firefox
firefox: /usr/bin/firefox /etc/firefox /usr/share/man/man1/firefox.1.gz
lori@ubuntu:~$
```

If you just want the path to the executable file to display and not display the path to the source file and the 'man' file, use the '-b' option. For example, the '*whereis -b firefox*' command will only display the result '*/usr/bin/firefox*'. This is very convenient because users will often search for more executable files than search for source and 'man' files of that program. Users can also search only for source files ('-s') or 'man' ('-m') files.

To find out the details of the command, type **man whereis** in the command **prompt** and press **Enter** .

## The difference between whereis and which

The whereis command shows the location of binary files, source files, and 'man' files with only one command, while the which command only shows the binary file location.

The whereis command finds from a special directory list for the binary file, the source file, and the 'man' file while the which command searches for the directories listed in the authorized PATH environment of the current user. With the whereis command, the special directory list can be found in the FILES section of the 'man' file.

By default, the whereis command displays all the results found while the which command only displays the first result found.

Because the whereis command only uses the *hard-coded path* to the command, you may not find the results you expect. If you are looking for a program that you think may have been installed in a directory that is not listed in the 'man' file of the whereis command, use the which command with the '-a' option to find all results from the system.

You finished reading the article "**Search for files and directories in Linux using the command line interface**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.

---