

Find out the locate command in Linux

On Linux systems, there are several commands that allow you to search for files, in which `find` and `locate` are the most used commands.

One of the most common activities when working on Linux is finding files and folders. On Linux systems, there are several commands that allow you to search for files, in which `find` and `locate` are the most used commands.

`locate` command is the fastest and easiest way to search for files, as well as folders by name.

In this tutorial, **TipsMake.com** will explain how to use the `locate` command in Linux.

How to use locate command in Linux

1. Locate settings (locate command not found)
 1. Install locate on Ubuntu and Debian
 2. Install locate on CentOS and Fedora
2. How to locate command
3. How to use the locate command

Locate settings (locate command not found)

Depending on the distribution and how the system is provided, `locate` packages can be preinstalled on Linux systems.

To check if the `locate` utility is installed on your system, open the terminal, type '`locate`' and press `Enter`. If the package is installed, the system will display the message "**locate: no pattern to search for specified**", otherwise you will see something like '**locate command not found**'.

If the `locate` not installed, you can easily install it with the distribution package manager.

Install locate on Ubuntu and Debian

```
$ sudo apt update $ sudo apt install mlocate
```

Install locate on CentOS and Fedora

```
$ sudo yum install mlocate
```

How to locate command

`locate` command searches for a certain pattern through the database file created by the `updatedb` command. The results are displayed on the screen, each result on one line.

During the installation of the `mlocate` package, a cron job is created to run the `updatedb` command every 24 hours. This ensures the database is updated regularly. For more information about cron jobs, check the file `/etc/cron.daily/mlocate`.

The database can be manually updated by running `updatedb` with root or user with `sudo` privileges:

```
$ sudo updatedb
```

The update process will take some time, depending on the number of files and folders, as well as your system speed.

Files created after updating the database will not be displayed in the `locate` command results.

Compared to the more powerful `find` command when searching for `locate` files and command systems that operate faster but lack many features and can only be searched by file name.

How to use the locate command

The syntax of the `locate` command is as follows:

```
locate [OPTION] PATTERN.
```

In the most basic form, when used without any options, the `locate` command will print the absolute path of all files and folders that match the search pattern and the user has the right to read the file in the link. search results.

For example, to search for a file called `.bashrc`, enter:

```
$ locate .bashrc
```

The output will include the names of all files containing the `.bashrc` string in their name:

```
/etc/bash.bashrc /etc/skel/.bashrc /home/linuxize/.bashrc /usr/share/base-files/
```

The `/root/.bashrc` file will not be displayed, because the example has run the command as a normal user, without access to the `/root` directory .

If the result list is long, for easier reading, you can move the output to `less` :

```
$ locate .bashrc | less
```

The `locate` command also accepts patterns that contain wildcards such as `*` . When patterns do not contain these characters, the command will search for `* PATTERN *` , which is why in the previous example, all files containing the search pattern in their names are displayed.

Wildcards are symbols used to represent zeros, one or more characters. For example, to search all **.md** files on the system you will use:

```
$ locate *.md
```

To limit search results, use the **-n** option followed by the number of results you want displayed. For example, the following command will search all **.py** files and display only 10 results:

```
$ locate -n 10 *.py
```

By default, `locate` commands perform a case-sensitive search. The **-i** (`--ignore-case`) option tells the `locate` command to run a search that is not case sensitive.

```
$ locate -i readme.md
```

Output is as follows:

```
/home/linuxize/p1/readme.md /home/linuxize/p2/README.md /home/linuxize/p3/ReadMe
```

To display the number of all matching items, use the **-c** (`--count`) option. The following command will return the number of all **.bashrc** files in their name:

```
$ locate -c .bashrc
```

Output:

```
6
```

By default, `locate` commands do not check if the files found still exist on the file system. If you deleted a file after updating the latest database and the file matches the search pattern, it will be included in the results.

To show only the names of files that still exist at the time the `locate` command is run, use the **-e** (`--existing`) option. For example, the following command will only return the existing **.json** files:

```
$ locate -e *.json
```

If you need to run a more complex search, you can use the **-r** (`--regex`) option, which allows you to search using regular expressions instead of patterns. This option can be specified multiple times.

For example, to search all **.mp4** and **.avi** files on your system, not case sensitive, you will run:

```
$ locate --regex -i "(.mp4|.avi)"
```

Locate command searches the file system for files and directories whose names match a given pattern. The command syntax is easy to remember and the results are displayed almost immediately.

For more information about all available options for the `locate` command, type `man locate` in the terminal.

If you have any questions, please leave a comment in the comment section below!

You finished reading the article "**Find out the locate command in Linux**" 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.
