



*Full credit is given to the above companies including the OS that this PDF file was generated!*

### ***Rocky Enterprise Linux 9.2 Manual Pages on command 'locate.1'***

#### ***\$ man locate.1***

locate(1)            General Commands Manual            locate(1)

#### NAME

locate - find files by name

#### SYNOPSIS

locate [OPTION]... PATTERN...

#### DESCRIPTION

locate reads one or more databases prepared by updatedb(8) and writes file names matching at least one of the PATTERNS to standard output, one per line.

If --regex is not specified, PATTERNS can contain globbing characters.

If any PATTERN contains no globbing characters, locate behaves as if the pattern were \*PATTERN\*.

By default, locate does not check whether files found in database still exist (but it does require all parent directories to exist if the database was built with --require-visibility no). locate can never report files created after the most recent update of the relevant database.

#### EXIT STATUS

locate exits with status 0 if any match was found or if locate was in?

voked with one of the --limit 0, --help, --statistics or --version options.

If no match was found or a fatal error was encountered, locate exits with status 1.

Errors encountered while reading a database are not fatal, search continues in other specified databases, if any.

## OPTIONS

-A, --all

Print only entries that match all PATTERNS instead of requiring only one of them to match.

-b, --basename

Match only the base name against the specified patterns. This is the opposite of --wholename.

-c, --count

Instead of writing file names on standard output, write the number of matching entries only.

-d, --database DBPATH

Replace the default database with DBPATH. DBPATH is a separated list of database file names. If more than one --database option is specified, the resulting path is a concatenation of the separate paths.

An empty database file name is replaced by the default database.

A database file name - refers to the standard input. Note that a database can be read from the standard input only once.

-e, --existing

Print only entries that refer to files existing at the time locate is run.

-L, --follow

When checking whether files exist (if the --existing option is specified), follow trailing symbolic links. This causes broken symbolic links to be omitted from the output.

This is the default behavior. The opposite can be specified using --nofollow.

-h, --help

Write a summary of the available options to standard output and exit successfully.

`-i, --ignore-case`

Ignore case distinctions when matching patterns.

`-l, --limit, -n LIMIT`

Exit successfully after finding LIMIT entries. If the `--count` option is specified, the resulting count is also limited to LIMIT.

`-m, --mmap`

Ignored, for compatibility with BSD and GNU locate.

`-P, --nofollow, -H`

When checking whether files exist (if the `--existing` option is specified), do not follow trailing symbolic links. This causes broken symbolic links to be reported like other files.

This is the opposite of `--follow`.

`-0, --null`

Separate the entries on output using the ASCII NUL character instead of writing each entry on a separate line. This option is designed for interoperability with the `--null` option of GNU `xargs(1)`.

`-S, --statistics`

Write statistics about each read database to standard output instead of searching for files and exit successfully.

`-q, --quiet`

Write no messages about errors encountered while reading and processing databases.

`-r, --regexp REGEXP`

Search for a basic regexp REGEXP. No PATTERNS are allowed if this option is used, but this option can be specified multiple times.

`--regex`

Interpret all PATTERNS as extended regexps.

`-s, --stdio`

Ignored, for compatibility with BSD and GNU locate.

`-V, --version`

Write information about the version and license of `locate` on standard output and exit successfully.

`-w, --wholename`

Match only the whole path name against the specified patterns.

This is the default behavior. The opposite can be specified using `--basename`.

## EXAMPLES

To search for a file named exactly `NAME` (not `*NAME*`), use

```
locate -b 'NAME'
```

Because `\` is a globbing character, this disables the implicit replacement of `NAME` by `*NAME*`.

## FILES

```
/var/lib/mlocate/mlocate.db
```

The database searched by default.

## ENVIRONMENT

`LOCATE_PATH`

Path to additional databases, added after the default database or the databases specified using the `--database` option.

## NOTES

The order in which the requested databases are processed is unspecified, which allows `locate` to reorder the database path for security reasons.

`locate` attempts to be compatible to `slocate` (without the options used for creating databases) and GNU `locate`, in that order. This is the reason for the impractical default `--follow` option and for the confusing set of `--regex` and `--regexp` options.

The short spelling of the `-r` option is incompatible to GNU `locate`, where it corresponds to the `--regex` option. Use the long option names to avoid confusion.

The `LOCATE_PATH` environment variable replaces the default database in BSD and GNU `locate`, but it is added to other databases in this implementation.

mentation and slocate.

AUTHOR

Miloslav Trmac <mitr@redhat.com>

SEE ALSO

updatedb(8)

mlocate

Sep 2012

locate(1)