

# Grep cheatsheet

This cheat sheet is intended to be a quick reminder for the main concepts involved in using the command line program grep and assumes you already understand its usage.

## # Getting Started

### Usage

Search standard output (i.e. a stream of text)

```
$ grep [options] search_string
```

Search for an exact string in file:

```
$ grep [options] search_string path/to/file
```

Print lines in myfile.txt containing the string "mellon"

```
$ grep 'mellon' myfile.txt
```

Wildcards are accepted in filename.

### Option examples

<code>-i</code>	<code>grep -i ^DA demo.txt</code>	Forgets about case sensitivity
<code>-w</code>	<code>grep -w "of" demo.txt</code>	Search only for the full word
<code>-A</code>	<code>grep -A 3 'Exception' error.log</code>	Display 3 lines after matching string
<code>-B</code>	<code>grep -B 4 'Exception' error.log</code>	Display 4 lines before matching string
<code>-C</code>	<code>grep -C 5 'Exception' error.log</code>	Display 5 lines around matching string
<code>-r</code>	<code>grep -r 'quickref.me' /var/log/nginx/</code>	Recursive search (within subdirs)
<code>-v</code>	<code>grep -v 'warning' /var/log/syslog</code>	Return all lines which don't match the pattern
<code>-e</code>	<code>grep -e '^al' filename</code>	Use regex (lines starting with 'al')
<code>-E</code>	<code>grep -E 'ja(s)cks)on' filename</code>	Extended regex (lines containing jason or jackson)
<code>-c</code>	<code>grep -c 'error' /var/log/syslog</code>	Count the number of matches
<code>-l</code>	<code>grep -l 'robot' /var/log/*</code>	Print the name of the file(s) of matches
<code>-o</code>	<code>grep -o search_string filename</code>	Only show the matching part of the string
<code>-n</code>	<code>grep -n "go" demo.txt</code>	Show the line numbers of the matches

## # Grep regular expressions

### Refer

- [Regex syntax](#) (quickref.me)
- [Regex examples](#) (quickref.me)

Please refer to the full version of the regex cheat sheet for more complex requirements.

### Wildcards

<code>.</code>	Any character.
<code>?</code>	Optional and can only occur once.
<code>*</code>	Optional and can occur more than once.
<code>+</code>	Required and can occur more than once.

### Quantifiers

<code>{n}</code>	Previous item appears exactly n times.
<code>{n,}</code>	Previous item appears n times or more.
<code>{,m}</code>	Previous item appears n times maximum.
<code>{n,m}</code>	Previous item appears between n and m times.

### POSIX

<code>[[:alpha:]]</code>	Any lower and upper case letter.
<code>[[:digit:]]</code>	Any number.
<code>[[:alnum:]]</code>	Any lower and upper case letter or digit.
<code>[[:space:]]</code>	Any whitespace.

### Character

<code>[A-Za-z]</code>	Any lower and upper case letter.
<code>[0-9]</code>	Any number.
<code>[0-9A-Za-z]</code>	Any lower and upper case letter or digit.

### Position

<code>^</code>	Beginning of line.
<code>\$</code>	End of line.
<code>^\$</code>	Empty line.
<code>\&lt;</code>	Start of word.
<code>\&gt;</code>	End of word.

### Related Cheatsheet

Awk Cheatsheet  
Quick Reference

Emacs Cheatsheet  
Quick Reference

Google Search Cheatsheet  
Quick Reference

Kubernetes Cheatsheet  
Quick Reference

Markdown Cheatsheet  
Quick Reference

Sed Cheatsheet  
Quick Reference

ES6 Cheatsheet  
Quick Reference

ASCII Code Cheatsheet  
Quick Reference

### Recent Cheatsheet



Share quick reference and cheat sheet for developers.

中文版 #Notes



Discover not just the hows, but also the whys of OAuth2 and OpenID Connect in this e-book!

cb

