

Bash Commands

<code>uname -a</code>	Show system and kernel
<code>head -n1 /etc/issue</code>	Show distribution
<code>mount</code>	Show mounted filesystems
<code>date</code>	Show system date
<code>uptime</code>	Show uptime
<code>whoami</code>	Show your username
<code>man <i>command</i></code>	Show manual for <i>command</i>

Bash Shortcuts

CTRL-c	Stop current command
CTRL-z	Sleep program
CTRL-a	Go to start of line
CTRL-e	Go to end of line
CTRL-u	Cut from start of line
CTRL-k	Cut to end of line
CTRL-r	Search history
!!	Repeat last command
<code>!<i>abc</i></code>	Run last command starting with <i>abc</i>
<code>!<i>abc</i>:p</code>	Print last command starting with <i>abc</i>
<code>!\$</code>	Last argument of previous command
ALT-.	Last argument of previous command
<code>!*</code>	All arguments of previous command
<code>^<i>abc</i>^123</code>	Run previous command, replacing <i>abc</i> with <i>123</i>

Bash Variables

<code>env</code>	Show environment variables
<code>echo \$NAME</code>	Output value of <i>NAME</i> variable

Bash Variables (cont)

<code>export NAME=value</code>	Set <i>NAME</i> to <i>value</i>
<code>\$PATH</code>	Executable search path
<code>\$HOME</code>	Home directory
<code>\$SHELL</code>	Current shell

IO Redirection

<code>cmd < file</code>	Input of <i>cmd</i> from <i>file</i>
<code>cmd1 <(cmd2)</code>	Output of <i>cmd2</i> as file input to <i>cmd1</i>
<code>cmd > file</code>	Standard output (stdout) of <i>cmd</i> to <i>file</i>
<code>cmd > /dev/null</code>	Discard stdout of <i>cmd</i>
<code>cmd >> file</code>	Append stdout to <i>file</i>
<code>cmd 2> file</code>	Error output (stderr) of <i>cmd</i> to <i>file</i>
<code>cmd 1>&2</code>	stdout to same place as stderr
<code>cmd 2>&1</code>	stderr to same place as stdout
<code>cmd &> file</code>	Every output of <i>cmd</i> to <i>file</i>
<i>cmd</i> refers to a command.	

Pipes

<code>cmd1 cmd2</code>	stdout of <i>cmd1</i> to <i>cmd2</i>
<code>cmd1 & cmd2</code>	stderr of <i>cmd1</i> to <i>cmd2</i>

Command Lists

<code>cmd1 ; cmd2</code>	Run <i>cmd1</i> then <i>cmd2</i>
<code>cmd1 && cmd2</code>	Run <i>cmd2</i> if <i>cmd1</i> is successful
<code>cmd1 cmd2</code>	Run <i>cmd2</i> if <i>cmd1</i> is not successful
<code>cmd &</code>	Run <i>cmd</i> in a subshell

Directory Operations

<code>pwd</code>	Show current directory
<code>mkdir dir</code>	Make directory <i>dir</i>
<code>cd dir</code>	Change directory to <i>dir</i>
<code>cd ..</code>	Go up a directory
<code>ls</code>	List files

Is Options

-a	Show all (including hidden)
-R	Recursive list
-r	Reverse order
-t	Sort by last modified
-S	Sort by file size
-l	Long listing format
-1	One file per line
-m	Comma-separated output
-Q	Quoted output

Search Files

<code>grep pattern files</code>	Search for <i>pattern</i> in <i>files</i>
<code>grep -i</code>	Case insensitive search
<code>grep -r</code>	Recursive search
<code>grep -v</code>	Inverted search
<code>grep -o</code>	Show matched part of file only
<code>find /dir/ -name name*</code>	Find files starting with <i>name</i> in <i>dir</i>



By **Dave Child** (DaveChild)
cheatography.com/davechild/
aloneonahill.com

Published 28th October, 2011.
 Last updated 29th February, 2020.
 Page 1 of 2.

Sponsored by **Readable.com**
 Measure your website readability!
<https://readable.com>

Search Files (cont)

`find /dir/ -user name` Find files owned by *name* in *dir*

`find /dir/ -mmin num` Find files modified less than *num* minutes ago in *dir*

`whereis command` Find binary / source / manual for *command*

`locate file` Find *file* (quick search of system index)

File Operations

`touch file1`
Create *file1*

`cat file1 file2`
Concatenate files and output

`less file1`
View and paginate *file1*

`file file1`
Get type of *file1*

`cp file1 file2`
Copy *file1* to *file2*

`mv file1 file2`
Move *file1* to *file2*

`rm file1`
Delete *file1*

`head file1`
Show first 10 lines of *file1*

`tail file1`
Show last 10 lines of *file1*

`tail -F file1`
Output last lines of *file1* as it changes

Watch a Command

`watch -n 5 'ntpq -p'`
Issue the 'ntpq -p' command every 5 seconds and display output

Process Management

`ps` Show snapshot of processes

`top` Show real time processes

`kill pid` Kill process with id *pid*

`pkill name` Kill process with name *name*

`killall name` Kill all processes with names beginning *name*

Nano Shortcuts

Files

Ctrl-R Read file

Ctrl-O Save file

Ctrl-X Close file

Cut and Paste

ALT-A Start marking text

CTRL-K Cut marked text or line

CTRL-U Paste text

Navigate File

ALT-/ End of file

CTRL-A Beginning of line

CTRL-E End of line

CTRL-C Show line number

CTRL-_ Go to line number

Search File

CTRL-W Find

ALT-W Find next

CTRL-\ Search and replace

More nano info at:
<http://www.nano-editor.org/docs.php>

Screen Shortcuts

`screen`
Start a screen session.

`screen -r`
Resume a screen session.

Screen Shortcuts (cont)

`screen -list`
Show your current screen sessions.

CTRL-A
Activate commands for screen.

CTRL-A c
Create a new instance of terminal.

CTRL-A n
Go to the next instance of terminal.

CTRL-A p
Go to the previous instance of terminal.

CTRL-A "
Show current instances of terminals.

CTRL-A A
Rename the current instance.

More screen info at:
<http://www.gnu.org/software/screen/>

File Permissions

`chmod 775 file`
Change mode of *file* to 775

`chmod -R 600 folder`
Recursively chmod *folder* to 600

`chown user.group file`
Change *file* owner to *user* and group to *group*

File Permission Numbers

First digit is owner permission, second is group and third is everyone.

Calculate permission digits by adding numbers below.

4 read (r)

2 write (w)

1 execute (x)



By **Dave Child** (DaveChild)
cheatography.com/davechild/
alnoneahill.com

Published 28th October, 2011.
Last updated 29th February, 2020.
Page 2 of 2.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>