

### Bash Commands

uname -a	Show system and kernel
head -n1 /etc/issue	Show distribution
mount	Show mounted filesystems
date	Show system date
uptime	Show uptime
whoami	Show your username
man <i>command</i>	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
! <i>abc</i>	Run last command starting with <i>abc</i>
! <i>abc</i> :p	Print last command starting with <i>abc</i>
!\$	Last argument of previous command
ALT-.	Last argument of previous command
!*	All arguments of previous command
^ <i>abc</i> ^123	Run previous command, replacing <i>abc</i> with 123

### Bash Variables

env	Show environment variables
echo \$ <i>NAME</i>	Output value of \$ <i>NAME</i> variable
export <i>NAME</i> = <i>value</i>	Set \$ <i>NAME</i> to <i>value</i>
\$PATH	Executable search path
\$HOME	Home directory
\$SHELL	Current shell

### IO Redirection

<i>cmd</i> < <i>file</i>	Input of <i>cmd</i> from <i>file</i>
<i>cmd</i> 1 <( <i>cmd</i> 2)	Output of <i>cmd</i> 2 as file input to <i>cmd</i> 1
<i>cmd</i> > <i>file</i>	Standard output (stdout) of <i>cmd</i> to <i>file</i>
<i>cmd</i> > /dev/null	Discard stdout of <i>cmd</i>
<i>cmd</i> >> <i>file</i>	Append stdout to <i>file</i>
<i>cmd</i> 2> <i>file</i>	Error output (stderr) of <i>cmd</i> to <i>file</i>
<i>cmd</i> 1>&2	stdout to same place as stderr

### Directory Operations

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

### ls 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

grep <i>pattern</i>	Search for <i>pattern</i> in <i>files</i>
grep -i	Case insensitive search
grep -r	Recursive search
grep -v	Inverted search
grep -o	Show matched part of file only
find / <i>dir</i> / -name <i>name</i> *	Find files starting with <i>name</i> in <i>dir</i>
find / <i>dir</i> / -user <i>name</i>	Find files owned by <i>name</i> in <i>dir</i>
find / <i>dir</i> / -mmin <i>num</i>	Find files modified less than <i>num</i> minutes ago in <i>dir</i>
whereis <i>command</i>	Find binary / source / manual for <i>command</i>
locate <i>file</i>	Find <i>file</i> (quick search of system index)

### File Operations

touch <i>file</i> 1	Create <i>file</i> 1
cat <i>file</i> 1 <i>file</i> 2	Concatenate files and output
less <i>file</i> 1	View and paginate <i>file</i> 1
file <i>file</i> 1	Get type of <i>file</i> 1
cp <i>file</i> 1 <i>file</i> 2	Copy <i>file</i> 1 to <i>file</i> 2
mv <i>file</i> 1 <i>file</i> 2	Move <i>file</i> 1 to <i>file</i> 2
rm <i>file</i> 1	Delete <i>file</i> 1
head <i>file</i> 1	Show first 10 lines of <i>file</i> 1

### Nano Shortcuts

<b>Files</b>	
Ctrl-R	Read file
Ctrl-O	Save file
Ctrl-X	Close file
<b>Cut and Paste</b>	
ALT-A	Start marking text
CTRL-K	Cut marked text or line
CTRL-U	Paste text
<b>Navigate File</b>	
ALT-/	End of file
CTRL-A	Beginning of line
CTRL-E	End of line
CTRL-C	Show line number
CTRL-_	Go to line number
<b>Search File</b>	
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 -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 <i>file</i>	Change mode of <i>file</i> to 775
chmod -R 600 <i>folder</i>	Recursively chmod <i>folder</i> to 600
chown <i>user</i> : <i>group</i> <i>file</i>	Change <i>file</i> owner to <i>user</i> and group to <i>group</i>

### File Permission Numbers

`cmd 2>&1`  
| stderr to same place as stdout

`cmd &> file`  
| Every output of `cmd` to `file`

`cmd` refers to a command.

### Pipes

`cmd1 | cmd2`  
| stdout of `cmd1` to `cmd2`

`cmd1 |& cmd2`  
| stderr of `cmd1` to `cmd2`

### Command Lists

`cmd1 ; cmd2`  
| Run `cmd1` then `cmd2`

`cmd1 && cmd2`  
| Run `cmd2` if `cmd1` is successful

`cmd1 || cmd2`  
| Run `cmd2` if `cmd1` is not successful

`cmd &`  
| Run `cmd` in a subshell

### Cheatographer



**Dave Child** (DaveChild)  
[cheatography.com/davechild/](http://cheatography.com/davechild/)  
[www.addedbytes.com](http://www.addedbytes.com)

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

`tail -f file1`  
| Output last lines of `file1` as it changes

### 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`

### Cheat Sheet

This cheat sheet was published on 28th October, 2011 and was last updated on 22nd November, 2011.

The first digit is the owner permission, the second the group and the third for everyone.

Calculate each of the three permission digits by adding the numeric values of the permissions below.

4	read (r)
2	write (w)
1	execute (x)

### Sponsor

**Envoy**, for simple and effective bug management.  
Try it free!  
<http://www.envoyapp.com>