

Stupid command line tricks

theScore[®]

Sportsball! Also eSports. And fantasy?

Shell/bash/zsh

emacs mode is default mode

Emacs bindings

ctrl-u – delete to beginning of line

ctrl-k – delete to end of line

ctrl-a – go to beginning of line

ctrl-e – go to end of line

ctrl-r – search through history (hit ctrl-r to keep going)

ctrl-w – delete word backward

Edit current line in your editor

```
export VISUAL=vim # nano, emacs, subl, etc.  
export EDITOR=vim  
ctrl-x e
```

vi mode for command line

Enable with: `set -o vi`

Turn off with: `set +o vi`

Works in bash, zsh, ksh, and more.

(pro tip: check out `set -o`)

vi bindings

esc - enter command mode

i - enter insert mode

u - undo

/ - search through history (n/N to cycle through matches)

j/k - go back/forward in history

b/w - to go back/forward a word

0/\$ - go to beginning/end of line

x - to delete a character

Directions work

d3w - delete 3 words

etc.

Edit current line in your editor

```
export VISUAL=vim # nano, emacs, subl, etc.
```

```
export EDITOR=vim
```

```
esc # enter command mode
```

```
v # edit with VISUAL/EDITOR
```

Command substitution

Command substitution with `` or '\$()':

old backtick style

```
rm `cat list_of_files_to_delete`
```

can nest with \$()

```
rm $(cat list_of_files_to_delete)
```

Piping command output

Pipe character takes input from other commands:

```
echo "Hello, world" | tr ' [a-z] ' ' [A-Z] '  
# HELLO, WORLD
```

Glob expansion

Renaming nested files without retyping out the path:

```
cp /home/foo/some_file.cpp /home/foo/some_file.cpp-old  
cp /home/foo/some_file.cpp{,-old}  
cp /home/foo/some_file.{cpp,cpp-old}
```

Go to home directory

No need for 'cd ~', just do this:

```
cd
```

The dash (-) character

The '-' character usually means use the previous thing:

`cd -` # go to previous directory

`git checkout -` # checkout previous branch

`git merge -` # merge previous branch into current branch

Git completion

zsh/bash can know about git if you want it to.

```
git checkout SPORTS-*tab*
```

→

```
git checkout SPORTS-123-my-branch-blah-blah-blah
```

Works for all other git commands (merge, branch, etc.)

Command repeat

'!!' repeats your last command:

```
touch /var/root/file_a
```

```
touch: /var/root/file_a: Permission denied
```

```
sudo !!
```


HEREDOC on command line

```
cat <<EOF  
Hello, world  
EOF  
# outputs 'Hello, world'
```

End standard-in with ctrl-d

Type this out on the command line:

```
cat <Enter>
```

```
Hello, world <Enter>
```

```
<ctrl-d>
```

Use **ctrl-d** to exit

Exits a lot of stuff (irb, any shell, pry prompts, etc.).

```
$ irb --simple-prompt
```

```
>> 1 + 1
```

```
=> 2
```

```
>> <ctrl-d>
```

```
$
```

Visual block select

In Terminal/iTerm:

⌘-alt click and drag

```

#( 08/21/15@ 2:44pm )( mcampbell@mbp ):~/my_new_rails_app@masterXXX
ls -l
total 40
-rw-r--r--  1 mcampbell  452276397  1476  24 Apr 14:33 Gemfile
-rw-r--r--  1 mcampbell  452276397  3828  24 Apr 14:34 Gemfile.lock
-rw-r--r--  1 mcampbell  452276397   478  24 Apr 14:33 README.rdoc
-rw-r--r--  1 mcampbell  452276397   249  24 Apr 14:33 Rakefile
drwxr-xr-x  8 mcampbell  452276397   272  24 Apr 14:33 app
drwxr-xr-x  7 mcampbell  452276397   238  24 Apr 14:34 bin
drwxr-xr-x 11 mcampbell  452276397   374  24 Apr 14:33 config
-rw-r--r--  1 mcampbell  452276397   153  24 Apr 14:33 config.ru
drwxr-xr-x  7 mcampbell  452276397   238  24 Apr 14:45 db
drwxr-xr-x  4 mcampbell  452276397   136  24 Apr 14:33 lib
drwxr-xr-x  5 mcampbell  452276397   170  24 Apr 14:44 log
drwxr-xr-x  7 mcampbell  452276397   238  24 Apr 14:33 public
drwxr-xr-x  9 mcampbell  452276397   306  24 Apr 14:33 test
drwxr-xr-x  3 mcampbell  452276397   102  24 Apr 14:33 tmp
drwxr-xr-x  3 mcampbell  452276397   102  24 Apr 14:33 vendor
#( 08/21/15@ 2:44pm )( mcampbell@mbp ):~/my_new_rails_app@masterXXX

```

Open output in editor

```
# vim  
ls -l | vim -
```

Tree of files

```
find .
```

Find files

```
find . | grep file_a
```

ssh tricks

ssh config can have aliases:

```
# ~/.ssh/config
Host host-a a-host my-super-alias-for-host-a
    Hostname foo-bar.compute-1.amazonaws.com
    User ubuntu
    ForwardX11 no
```


Use the alias

```
ssh host-a # ssh to host-a
```

```
scp host-a:/tmp/file_a.txt . # scp from host-a
```

```
ssh host-a ls -l /tmp/file_a.txt # run command on host-a
```

Fin

Thanks!

Sometimes I blog: markcampbell.me

Twitter: @Nitrodust