

Vim Survival Guide

Grant McLean

<grant@catalyst.net.nz>

What is Vim?



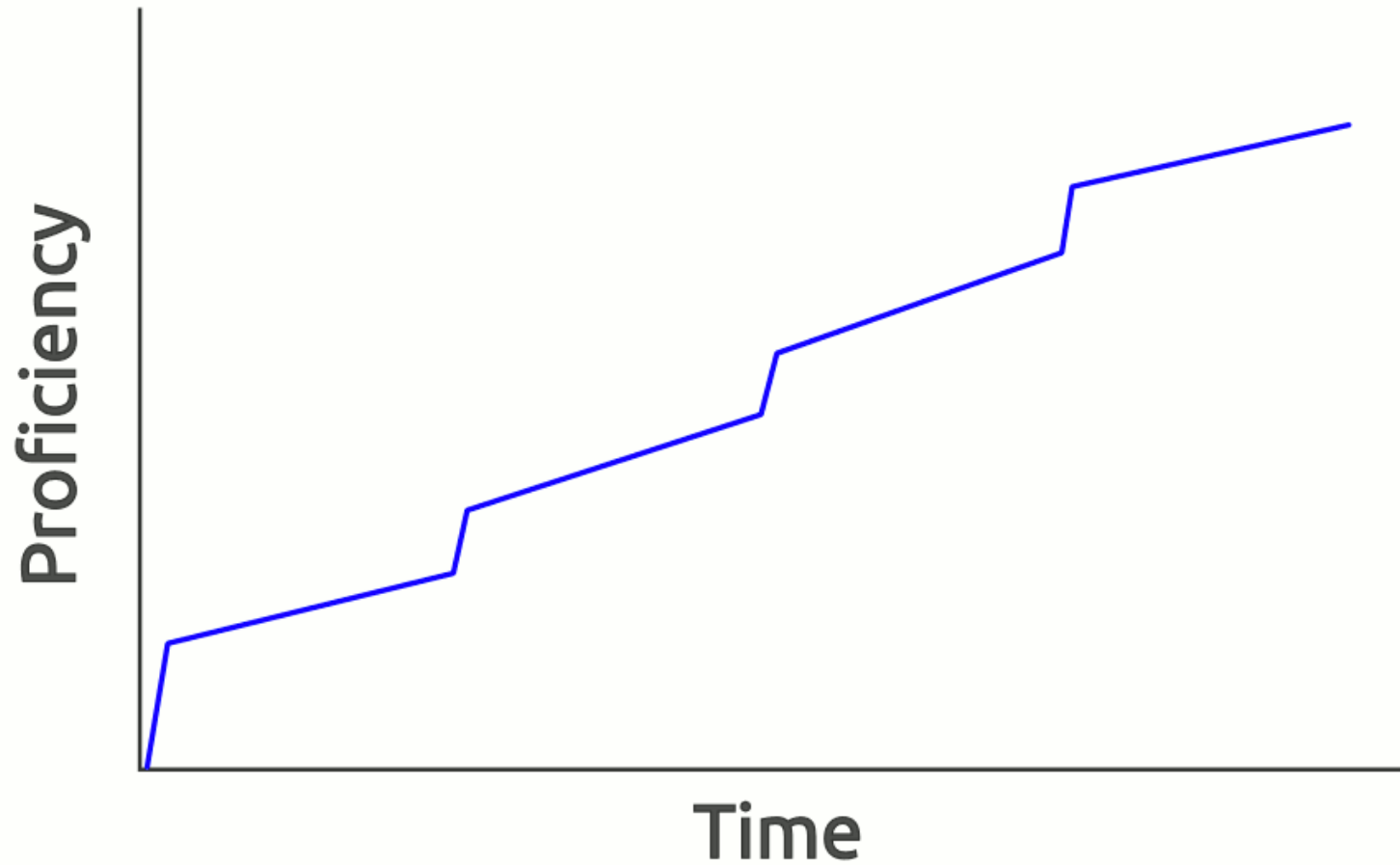
- **Vim = vi improved**

Types of Vim users

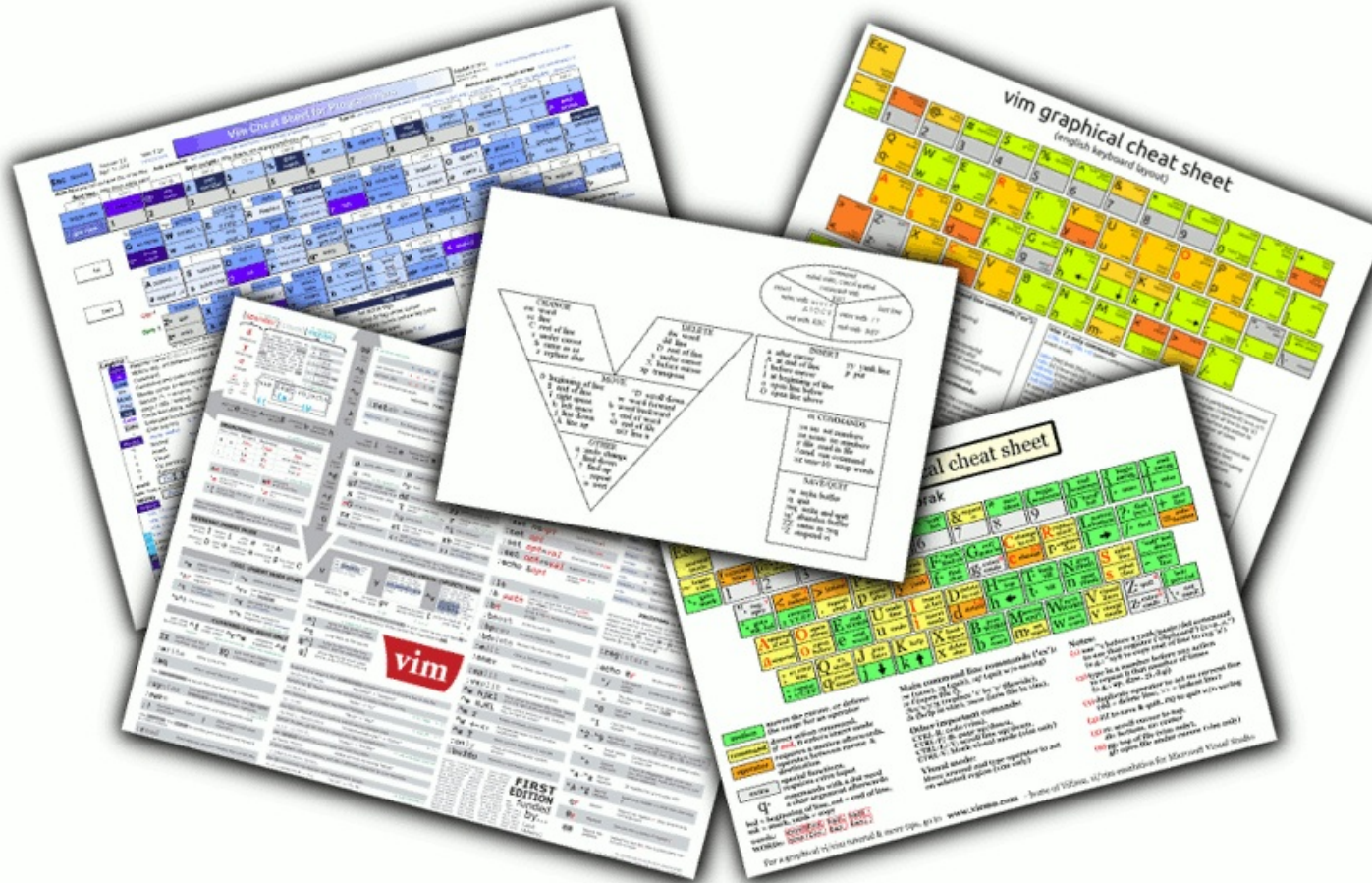


- **Vim experts**
- **Vim journeypersons**
- **Vim prisoners**
- **Vim curious**
- **Vim haters**

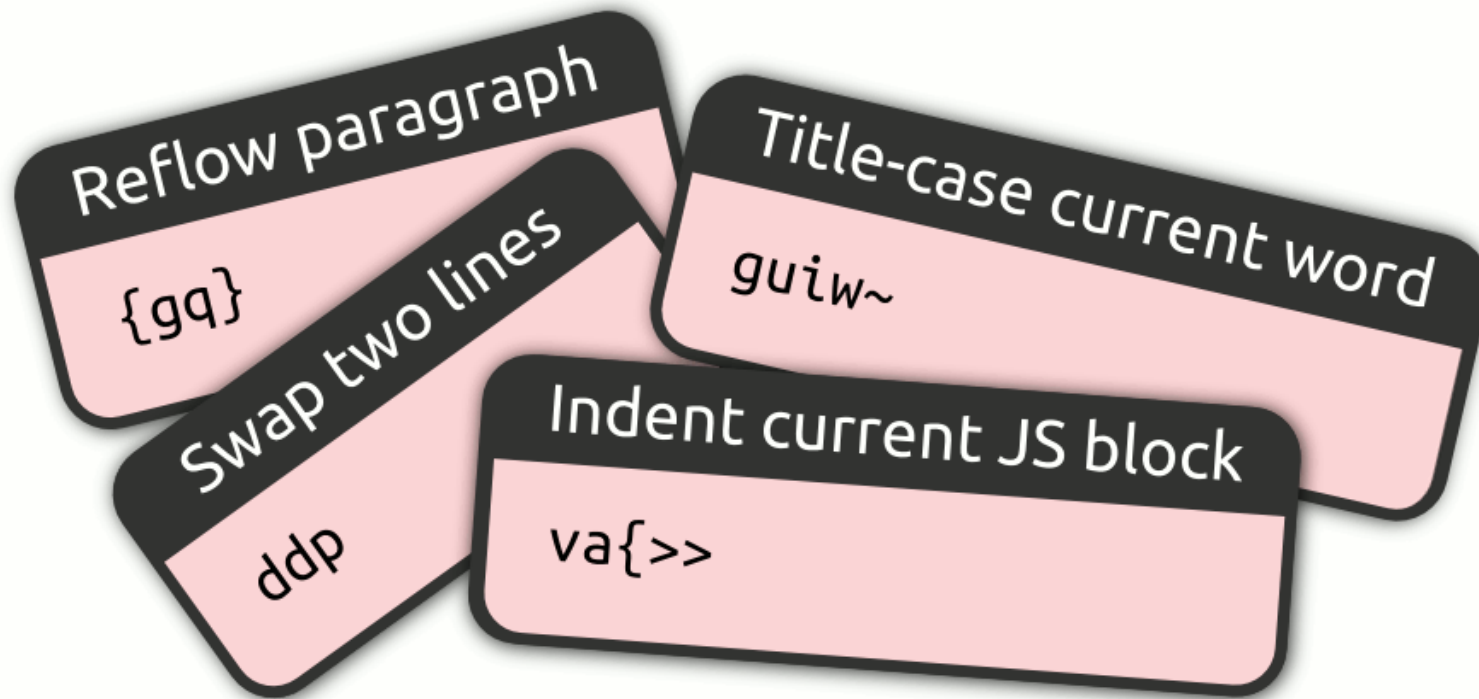
Vim learning curve



Vim cheat-sheets



Vim recipes



Targets for today



- **Comfort level**
- **Understanding “the Vim way”**

Getting in to Vim

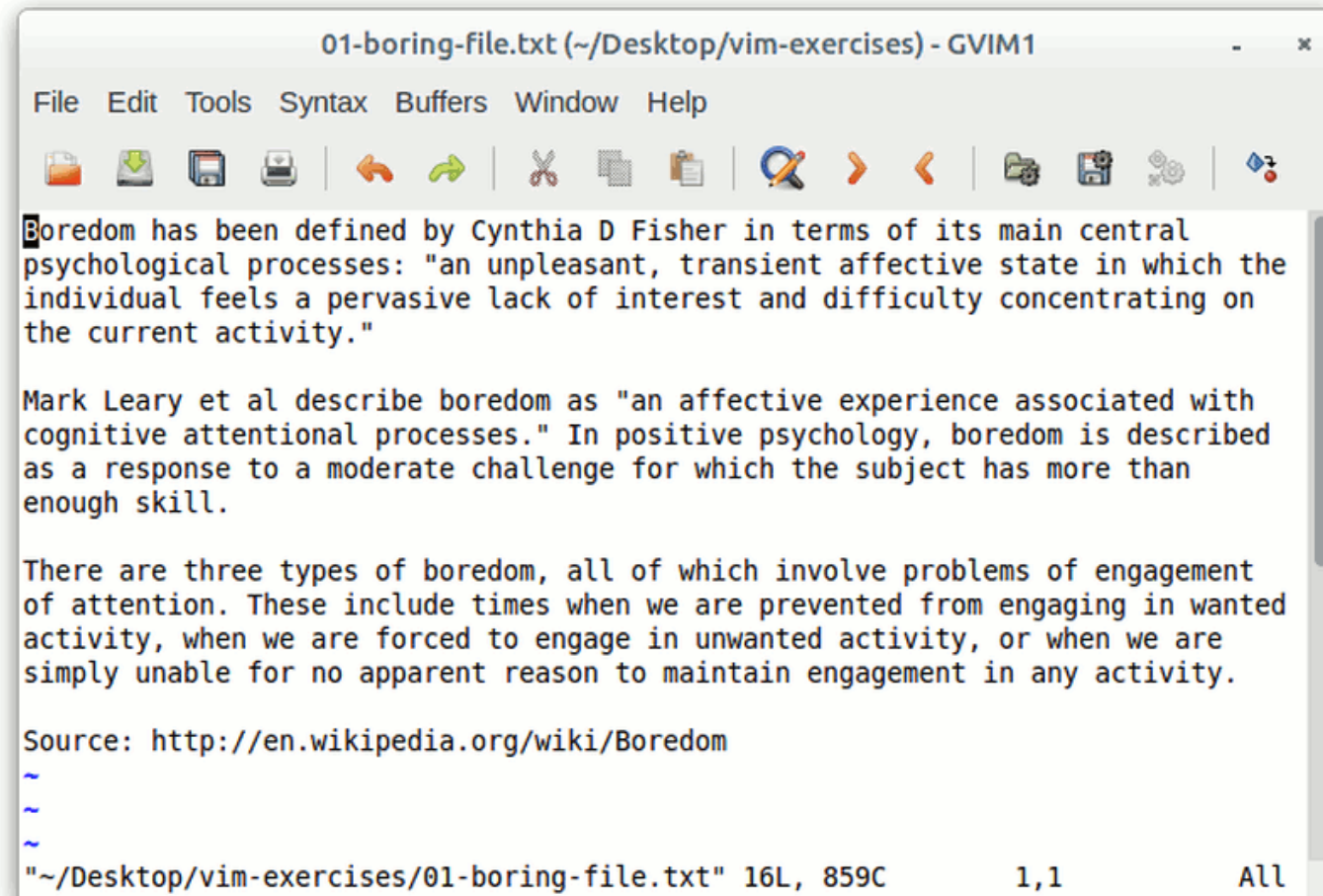


- **From the command-line**

```
$ vim 01-boring-file.txt
$ vi 01-boring-file.txt
$ gvim 01-boring-file.txt
$ alias vi="gvim"
```

- **Right click file ► Open with GVim**

GVim window



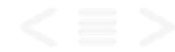
```
01-boring-file.txt (~/Desktop/vim-exercises) - GVIM1
File Edit Tools Syntax Buffers Window Help
[Icons]
Boredom has been defined by Cynthia D Fisher in terms of its main central
psychological processes: "an unpleasant, transient affective state in which the
individual feels a pervasive lack of interest and difficulty concentrating on
the current activity."

Mark Leary et al describe boredom as "an affective experience associated with
cognitive attentional processes." In positive psychology, boredom is described
as a response to a moderate challenge for which the subject has more than
enough skill.

There are three types of boredom, all of which involve problems of engagement
of attention. These include times when we are prevented from engaging in wanted
activity, when we are forced to engage in unwanted activity, or when we are
simply unable for no apparent reason to maintain engagement in any activity.

Source: http://en.wikipedia.org/wiki/Boredom
~
~
~
"~/Desktop/vim-exercises/01-boring-file.txt" 16L, 859C      1,1      All
```

Getting out of Vim



```
:wq<Enter>    # Write and Quit  
:w<Enter>     # Write, don't Quit  
:q<Enter>     # Quit don't Write  
:q!<Enter>    # Quit even if changes
```

```
ZZ            # Write and Quit
```

```
:x<Enter>     # Write and Quit
```

- **GVim – just close the window**



Deleting stuff



- **Delete one character**

x

- **Delete one line**

dd

- **Repeat the last thing**

.

- **Undo the last thing**

u

Undo/redo



- Undo the last thing

`u`

- Redo the thing you just 'undid'

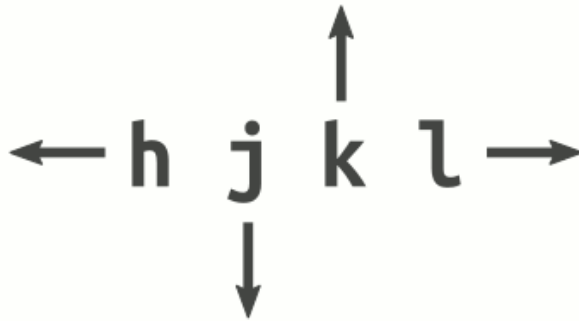
`<C-r>`

Moving around the file

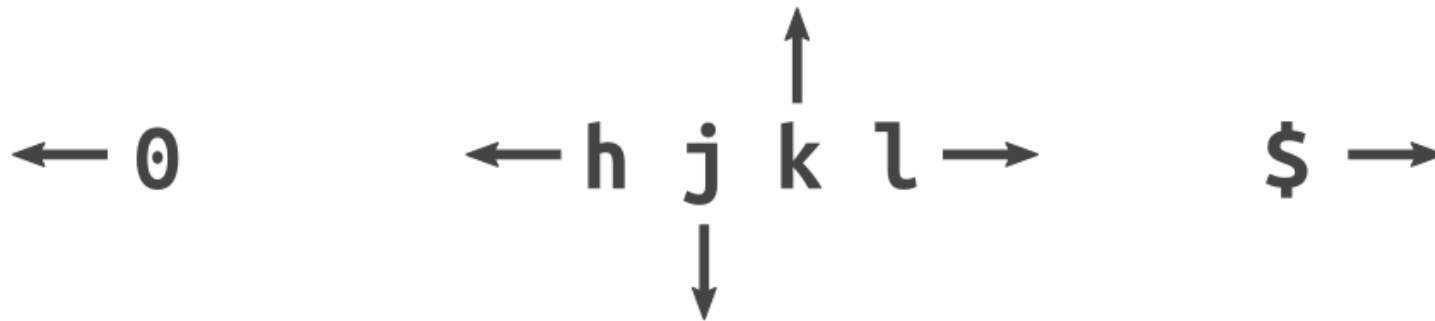


h j k l

Moving around the file



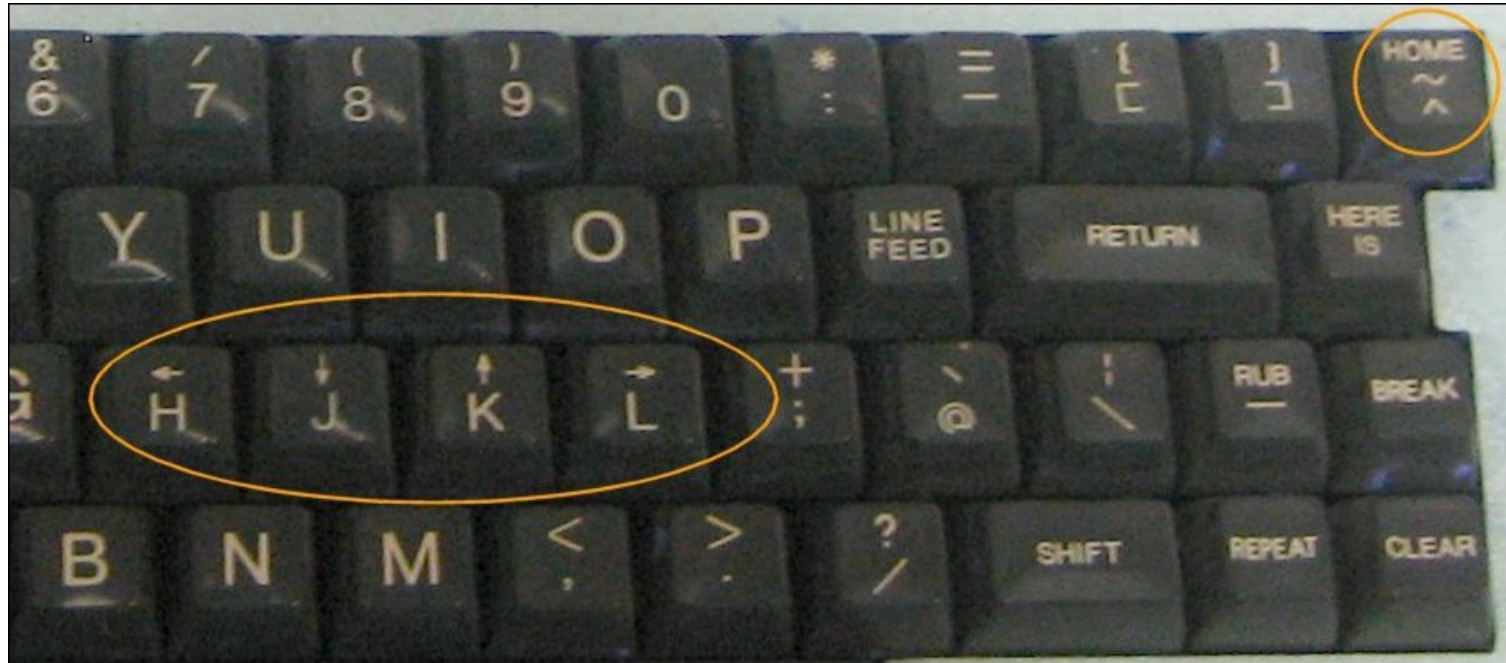
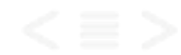
Moving around the file



Why H J K L ?



Why H J K L ?



Inserting text



- **Insert before current character**

`i # <Esc> when done`

- **Append after current character**

`a`

- **Append to end of line**

`A`

- **Insert at beginning of line**

`I`

Inserting text



- **Append some text**

`A<Space>Blah<Esc>`

- **Repeat last action**

.

Inserting new lines



- **Open a new line below**
 -
- **Open a new line above**
 -

Vim modes



- **Normal mode**
- **Insert mode**
- **Append mode**
- **Replace mode**
- **Visual mode**
- **Command-line mode**

More motion commands



- **Move to start of next word**

w

- **Move to end of current word**

e

- **Move to beginning of word**

b

Repetition prefix



- **Delete 1 line of text**

`dd`

- **Delete 9 lines of text**

`9dd`

- **Move down 5 lines**

`5j`

- **Move forward 3 words**

`3w`

Repetition prefix



- What does this do?

`3iNom <Esc>`

`Nom Nom Nom`

- And this?

`78i=<Esc>`

=====

Composing commands



- How to replace a word?
- Delete the word, then insert

`dwi`

- or 'change' the word

`cw`

- or 'change' the next 3 words

`3cw`

More motion commands



- **Move forward to the next 'e'**

`fe`

- **Delete up to the next 'e'**

`dfe`

- **Change up to the next '.'**

`cf.`

- **Change to the end of the sentence**

`c)`

More movement commands



- **Move to the next blank line**

}

- **Move to the previous blank line**

{

- **Delete the current paragraph**

{d}

More 'finding' commands



- Forward to the next 'x'

`fx`

- Back to the previous 'x'

`Fx`

- Forward to before the next 'x'

`tx`

- Back to before the previous 'x'

`Tx`

More editing commands



```
<div class="navbar">
```

- Move to the start of 'navbar'

```
fn
```

- Change up until the closing quote

```
ct "
```

- Change in the quotes

```
ci "
```

More editing commands



- **Change inside the quotes**

```
ci "header<Esc>
```

- **Change inside *and* the quotes**

```
ca " 'header' <Esc>
```

- **Change inside parentheses**

```
ci (
```

- **Change inside angle brackets**

```
ci<
```

Getting around a file



- **Go to line 12**

`12G`

- **Go to the last line**

`G`

- **Go to the first line**

`gg`

Cut/paste



- **Delete the current line**

`dd`

- **Paste below the current line**

`p`

- **Paste above the current line**

`P`

Copy/paste



- **'Yank' the current line**

`yy`

- **Paste below the current line**

`p`

- **Yank to the next blank line**

`y}`

Visual selections



- **Enable visual selection mode**

<code>v</code>	<code># Select range by character</code>
<code>V</code>	<code># Select range by line</code>
<code><C-v></code>	<code># Visual block mode</code>

- **Use motion keys to extend selection**
- **GVim: click and drag with the mouse**

Visual selections



- **Delete selection**

`{Visual}x`

- **Yank (copy) selection**

`{Visual}y`

- **Indent selection**

`{Visual}>`

Searching



- **Search for 'monkey'**

`/monkey<Enter>`

- **Find the next occurrence**

`n`

- **Find the previous occurrence**

`N`

Searching



- **Search for word under cursor**

- **Search backwards for word under cursor**

#

Search options



- **Enable search match highlighting**

```
:set hlsearch
```

- **Make searches case-insensitive**

```
:set ignorecase
```

- **Enable incremental searching**

```
:set incsearch
```

\$HOME/.vimrc



```
set autoindent
set shiftwidth=2
set softtabstop=2
set expandtab
set ignorecase
set smartcase
set nowrapscan
set incsearch
set nobackup
```

Command mode



- **You have already seen:**

`:wq`

- **Write to a different filename**

`:w new-file-name`

Command mode



- Working with ranges

<code>:d</code>	<code># Delete current line</code>
<code>:1,10d</code>	<code># Delete first 10 lines</code>
<code>:-3,+3d</code>	<code># Delete 7 lines around current</code>
<code>:\$d</code>	<code># Delete to end of file</code>
<code>:g/pattern/d</code>	<code># Delete every matching line</code>

Command mode



- **Search and replace**

`:s/old/new/` # First occurrence on line

`:s/old/new/g` # All occurrences on line

`:1,$s/old/new/g` # All occurrences in file

`:%s/old/new/g` # All occurrences in file

`{Visual}s/old/new/g` # All in selection

Getting help



`:help`

`:h`

`:h x`

`# Help about x`

`:h :x`

`# Help about :x`

`<F1>`

`# Intro to help system`

`:q`

`# Close help window`

Getting help



- **Follow a help link ('tag')**

`<C-] >`

- **'Jump' back**

`<C-T>`

- **Finding help topics**

`:h topic<C-D>`

The End



- Questions?
- <http://vim-adventures.com/>