



Using Vim

Justin Chao

Technology Development Program - Class 11

November 6, 2017

Introduction

Introduction

- Introduction
- Basics
- Getting Stuff Done
- Advanced Topics

Vim is a highly configurable, modal text editor built to make creating and changing any kind of text very efficient.

- System administration
- Programming
- Working with HTML, LaTeX, or other markup languages
- Heavy editing of plain text files



Introduction

- It's ubiquitous.

It's installed on every Unix-like system, and it's free!

- It's scalable.

You can use it just to edit config files or it can become your entire writing platform.

- It's powerful.

Because it works like a language vim takes you from frustrated to demigod very quickly.

Introduction

- Level 0: knows nothing about vim
- Level 1: knows vim basics
- Level 2: knows visual mode
- Level 3: knows various motions
- Level 4: not needing visual mode
- Level 5: magic things are happening on the screen

Basics

Basics

Verbs	d	delete
	c	change
	y	yank (copy)
	v	visually select
Modifiers	i	inside
	a	around
	NUM	number (1,2,10)
	t	searches for something and stops before it
	f	searches for that thing and lands on it
Nouns	w	word
	s	sentence
	p	paragraph
	t	tag (think HTML/XML)
	b	block (think programming)

Verb - Modifier - Noun

delete two words

d2w

change inside sentence

cis

yank (copy) inside paragraph

yip

change to the < character

ct<

Getting Stuff Done

Getting Stuff Done

<code>vi/vim file</code>		open your file in vim
<code>:w</code>		write your changes to the file
<code>:q!</code>		quit vim without saving your changes
<code>:wq</code>		write your changes, and quit

Getting Stuff Done

j	move down one line
k	move up one line
h	move left one character
l	move right one character
gg	go to the top of the file
G	go to the bottom of the file

Getting Stuff Done

0	move to the beginning of the line
\$	move to the end of the line
^	move to the first non-blank character in the line
t"	jump to right before the next quotes
f"	jump and land on the next quotes

Getting Stuff Done

w		move forward one word
b		move back one word
e		move to the end of your word

Getting Stuff Done

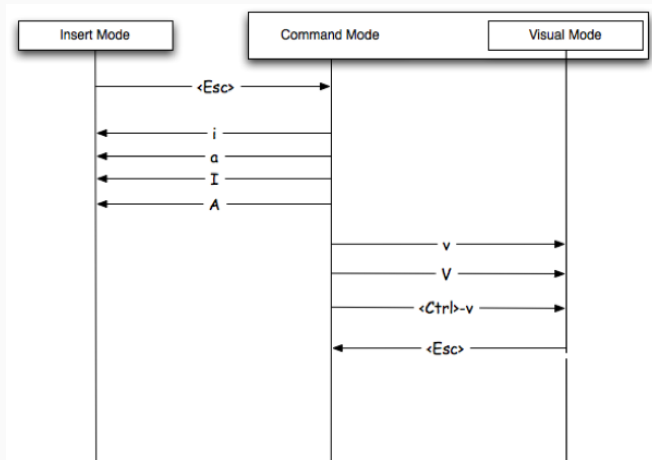
```
# search for include  
/include<CR>
```

```
# jump forward and land on the < character  
f<
```

```
# jump forward and land right before the <  
character  
t<
```

```
# change to and including the < character  
ct<
```


Getting Stuff Done



Getting Stuff Done

From Command/Normal mode:

- | | | |
|---|--|---|
| i | | insert before the cursor |
| I | | insert at the beginning of the line |
| a | | append after the cursor |
| A | | append at the end of the line |
| o | | open a new line below the current one |
| O | | open a new line above the current one |
| r | | replace the one character under your cursor |
| R | | replace the character, but stay in insert mode |
| S | | substitute the entire current line |
| C | | delete the line from where you're at, and enter insert mode |

Getting Stuff Done

From Command/Normal mode:

x	exterminate (delete) the character under the cursor
X	exterminate (delete) the character before the cursor
dd	delete the current line
dt.	delete from where you are to the period
D	delete to the end of the line
J	join the current line with the next one (delete what's between)

Getting Stuff Done

u		undo your last action
Ctrl-r		redo the last action

Advanced Topics

`" "`

Given a discreet movement action, combine it with a repeatable command captured in the `"."`.

Advanced Topics

change "foo" to "bar" on every line

:%s/foo/bar/g

change "foo" to "bar" on just the current line

:s/foo/bar/g

word text objects

double quotes

single quotes

parenthesis

paragraphs

back tics

brackets

braces

tags

Advanced Topics

Visual mode magnifies the power of everything we've learned so far, by allowing you to apply commands to the text that's currently highlighted.

v	character based
V	lined based
vi(select inside of parenthesis
vi[select inside of brackets
Ctrl-v	column based

qa	start recording a macro named "a"
q	stop recording
@a	play back the macro

1. Search within the line for "widget"
2. Go to the end of the word and add "-maker"
3. Go to the beginning of the line and add a colon
4. Go to the end of the line and add a period
5. Delete any empty spaces at the end of the line

Advanced Topics

tpope/vim-surround	all about the "surroundings"!
tpope/vim-fugitive	a Git wrapper
scrooloose/nerdcommenter	easy commenting
scrooloose/nerdtree	a tree explorer plugin
junegunn/fzf	a general-purpose fuzzy finder

creating and modifying your .vimrc file

Resources

Additional Resources

<code>vimtutor</code>	built-in tutor to get your started
<code>vim-adventures.com</code>	game based tutorial
<code>github.com/j-chao/dotfiles</code>	my <code>.vimrc</code>