

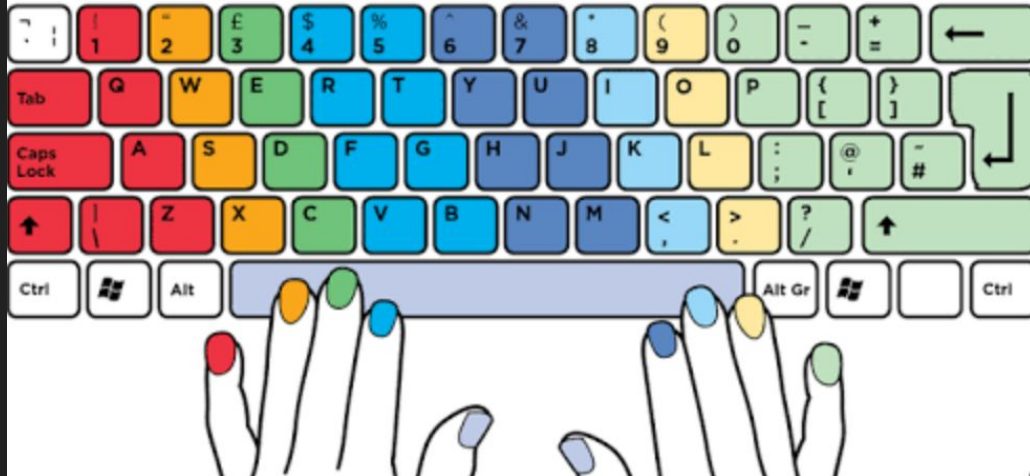


Code Blazingly Fast  with
Neovim

JUSTIN EDGAR

Prerequisites

- Have Git installed
- Have Neovim installed (latest version)
- Know how to Touch-Type (learn this first)



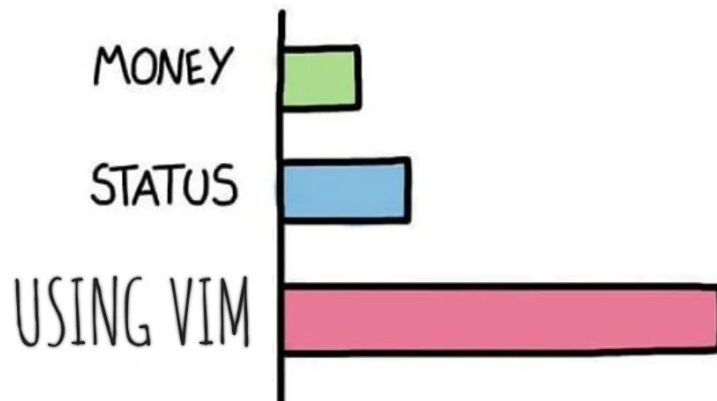
Slides + Demo + Exercises

```
$ git clone https://github.com/Heaveil/Vim-Talk.git
```

Outline

- Introduction
 - What is Vim and Neovim
 - Neovim and The Terminal
 - Caveats
 - Prerequisites
- Basics
 - Modes
 - Navigation
 - Motions
 - Macros
- Config & Plugin Basics
- Q & A

WHAT GIVES PEOPLE FEELINGS OF POWER



version 1.1
April 1st, 06

Vim graphical cheat sheet

Esc
normal mode

| | | | | | | | | | | | |
|---------------|-----------------|--------------|--------------|--------|--------------|--------|--------------|------------------|----------------|-----------------|--------------|
| ~ toggle case | external filter | @ play macro | # prev ident | \$ eol | % goto match | repeat | * next ident | (begin sentence |) end sentence | "soft" bol down | + next line |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 "hard" bol | - prev line | = autoformat |

| | | | | | | | | | | | |
|----------------|-------------|------------|----------------|-------------|-------------|-------------|---------------|----------------|----------------|---|------------|
| Q ex mode | W next WORD | E end WORD | R replace mode | F back till | Y yank line | U undo line | I insert line | P paste before | { begin parag. | } | end parag. |
| q record macro | w next word | e end word | r replace char | f "till | y yank | u undo | i insert word | P paste after | [misc |] | misc |

| | | | | | | | | | | |
|-----------------|--------------|-----------------|-------------------|------------------|--------------|--------------|--------|-----------------|--------|---------------|
| A append at eol | S subst line | D delete to eol | F "back find char | G eof/ to bottom | H screen top | J join lines | K help | L screen bottom | repeat | bol/ goto col |
| a append | s subst char | d delete | f, find char | g, extra c | h ← | j ↓ | k ↑ | l → | repeat | bol/ goto col |

| | | | | | | | | | |
|-----------|---------------|-----------------|----------------|-------------|---------------|--------------|-------------|--------------|--------|
| Z quit | X backspace | C change to col | V visual lines | B prev WORD | N prev (find) | M screen mid | < un-indent | > indent | ? find |
| Z extra c | X delete char | C change | V visual mode | b prev word | n next (find) | m set mark | < reverse | > repeat cmd | / find |

motion moves the cursor, or defines the range for an operator

command direct action command, if red, it enters insert mode

operator requires a motion afterwards, operates between cursor & destination

extra special functions, requires extra input commands with a dot need a char argument afterwards

q bol = beginning of line, eol = end of line, mk = mark, yank = copy

WORDS: `quux[foo] bar` `quux[foo] bar`

Main command line commands ('ex'):

- rw (save), q (quit), w! (quit w/o saving)
- zf (open file), :ls/x/y/p (replace 'x' by 'y' filewide), :h (help in vim), :new (new file in vim),

Other important commands:

- CTRL-R: redo (vim), CTRL-F/B: page up/down, CTRL-E/Y: scroll line up/down, CTRL-V: block-visual mode (vim only)

Visual mode:

- Move around and type on selected region (vim only)

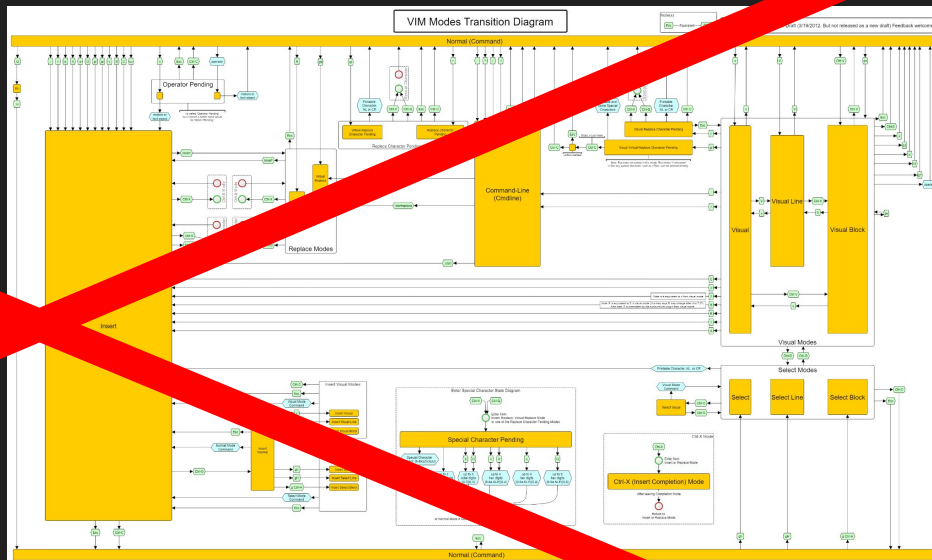
Notes:

- (1) use 'x' before a yank/paste/delete to use that register (clipboard) (e.g.: "ay8 to copy register 'a')
- (2) type in a number before an action to repeat it 'x' of times (e.g.: "x8 to delete 8 lines)
- (3) use 'g' before an operator to act on current line to repeat it 'x' of times (e.g.: "g8 to delete line, >> = indent line)
- (4) to save & quit, ZZ to quit w/o saving
- (5) zt: scroll cursor to top, zb: bottom, zc: center
- (6) gg: top of file (vim only), gG: open file under cursor (vim only)

VIM Modes Transition Diagram

The diagram illustrates the transitions between Vim modes: Normal (Command), Insert, and Visual (Visual Line, Visual Block). It shows how various keys and commands trigger mode changes, such as pressing 'i' to enter Insert mode or 'v' to enter Visual mode. The diagram also includes a section for 'Special Character Pending' and a 'Quit-A (Short) Completion Mode'.

For a graphical vi/vim tutorial & more tips, go to www.vim-emulation.com - home of ViEmu, vi/vim emulation for Microsoft Visual Studio



Goal

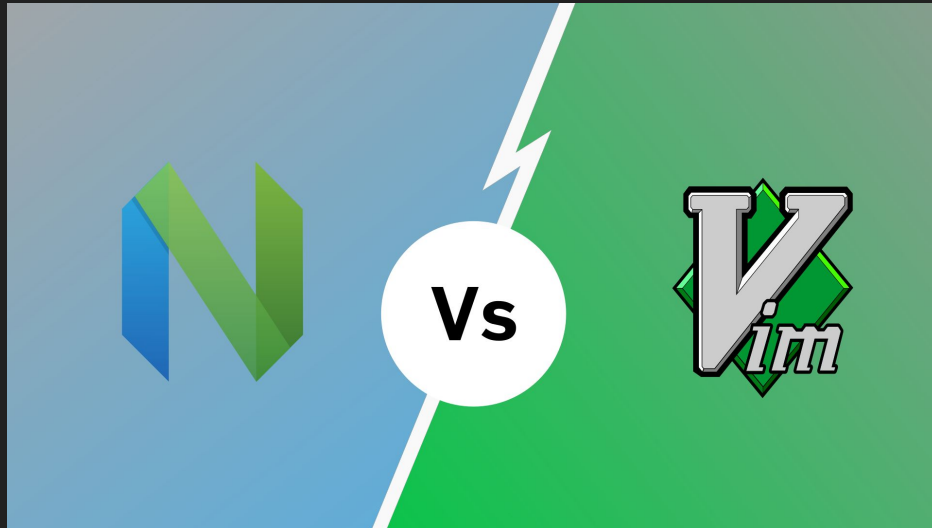


Goal



What is Vim and Neovim

- Vim is a text editor (it edits text)
- Neovim is a community fork of Vim



Neovim and The Terminal

- Neovim \neq Terminal
- Neovim is a program that lives inside the terminal
- The philosophy is to never leave your terminal



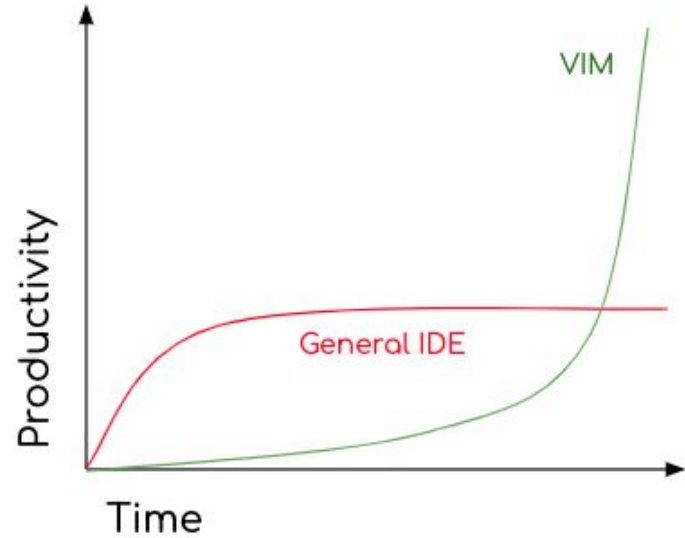
Why Neovim

- Mouseless Development
- Life Long Skill
- Super Efficient
- Blazingly Fast 🚀
- Vast Plugin Ecosystem
- Makes Coding Super Fun
- Bragging Rights (I Use Neovim Btw)



Caveats

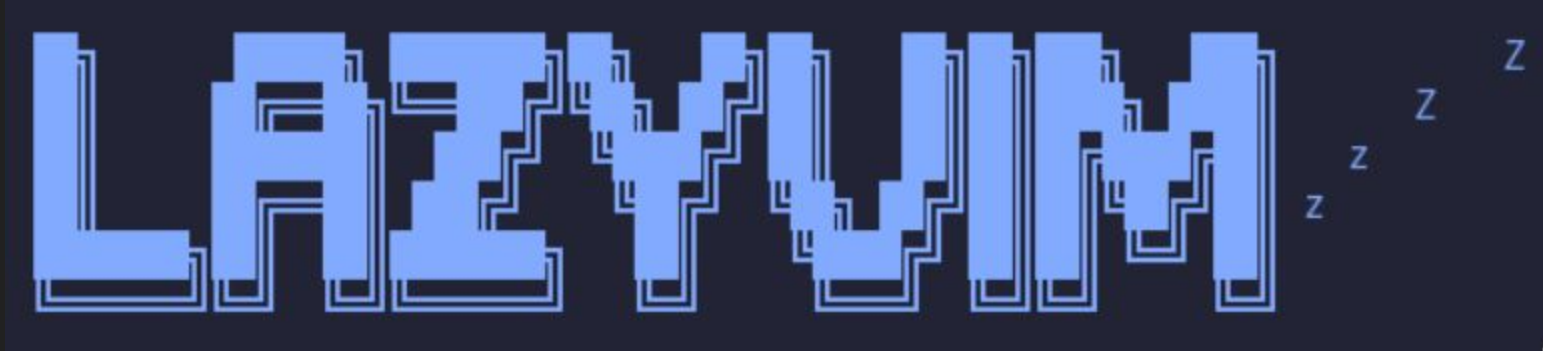
- Steep Learning Curve
- Configuration Hell



Getting Started

Kickstart: <https://github.com/nvim-lua/kickstart.nvim>

Lazyvim: <https://www.lazyvim.org/>





Vim v1.27.3

vscodevim



6,512,210



(368)

Vim emulation for Visual Studio Code

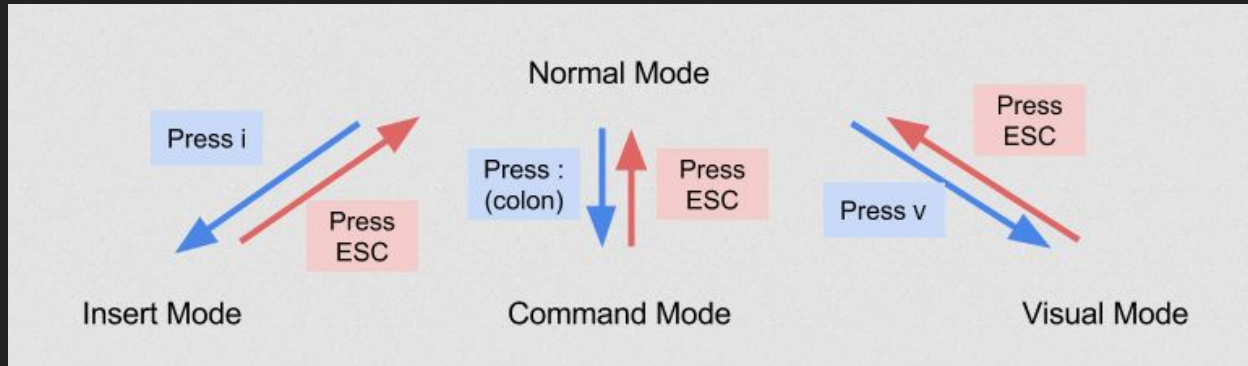
Install



Let's Start!

Vim Modes

- Vim is a mode based editor. It has 4 main modes (Normal, Insert, Visual, Command)
- Press **esc** on any mode to go into normal mode, normal mode should be your default
- You can check out **Vimtutor** to learn the basics of vim and use **:help** to go into the friendly manual



Basic Navigation

** To move the cursor, press the h,j,k,l keys as indicated. **

^
k
< h l >
j
v

Hint: The h key is at the left and moves left.
The l key is at the right and moves right.
The j key looks like a down arrow.

Horizontal Navigation

- word
- end of word
- back
- ^ - first character
- \$ - last character
- find / Find <character>
- till / Till <character>

Vertical Navigation

- ctrl-up - moves cursor up ½ page
- ctrl-down - moves cursor down ½ page
- ctrl-original - moves cursor to previous position
- **gg** - moves cursor to start of file
- **G** - moves cursor to end of file

RELATIVE LINE NUMBERS

<number><j/k>

Insert Mode

- insert - before cursor
- Insert - start of line
- append - after cursor
- Append - end of line
- open - new line below cursor
- Open - new line above cursor

Visual Mode

Visual Line Mode - **V**

Tab left - **<**

Tab Right - **>**

Opposite End - **o**

Visual Block Mode - **ctrl-v**

Increment - **ctrl-a**

Global Increment - **g ctrl-a**

More Actions

- ddelete
- undo
- ctrl-redo
- yank
- paste
- replace - replace a character
- Replace - go to replace mode
- / - go to find mode
- next - find next occurrence
- Next - find previous occurrence

Command Mode

- Press colon ":" while in normal mode to go to command mode
- In command mode
 - **w** - write/save
 - **q** - quit
 - **wq** - write/save and quit
 - **q!** - quit and not save
 - **e** <file-name> - edit a file (creates one if it does not exist)
 - <line-number> - goes to the line number (useful for debugging)

Anatomy of Vim Motions

<verb><modifier><noun>

Verbs

- visual
- deleate
- change
- yank

Modifiers

- Numbers
- till
- find
- inside
- around

Nouns

- hjkl
- word
- () [] { }
- “” ‘ ’

Find and Replace (Substitute)

:<range>s/<search pattern>/<replace string>/<options>

:%s/hello/world/g → replaces every “hello” with “world”

Macros

- Macros are sequences of motions squished together
- To start recording a macro, press q with a key (example j)
- To end the macro recording, simply press q again
- To apply the macro, press @<key-used-to-record> (e.g. @j)
- You can type numbers before applying the macro to apply the macro n times. <number>@<key-used-to-record> (e.g. 5@j)

Config & Plugins



mason.nvim



Harpoon Man

SNL
LIVE FROM NEW YORK



• Neovim motions on speed! •

Plugin Repos

- <https://github.com/nvim-telescope/telescope.nvim>
- <https://github.com/ThePrimeagen/harpoon/tree/harpoon2>
- <https://github.com/smoka7/hop.nvim>
- <https://github.com/nvim-neo-tree/neo-tree.nvim>
- <https://github.com/lewis6991/gitsigns.nvim>
- <https://github.com/nvim-lualine/lualine.nvim>
- <https://github.com/folke/noice.nvim>

Complimentary Tools

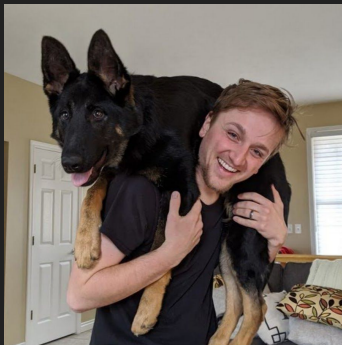


tmux

Resources



ThePrimeagen



TJ DeVries



Typecraft

What's Next?

- If you like what you saw today, then try out Neovim. Maybe for a week or so. I know the first few weeks will be very challenging, but I guarantee once you get a hang of it, coding will be a much more enjoyable experience.
- If you only like the motion or not fully convinced yet. You can start with the vim plugin for your favorite editor. You don't have to jump into Neovim head first. You can take it bit by bit.
- If you don't like what you saw today, that's okay. At least, you are willing to learn something new.

Tips for learning the motions

- Start with **hjkl** for navigation
- Use **a** to go to insert mode
- Use **v** and **V** to go to visual mode
- Use **d y p** to delete yank paste
- Start from these basic motions and incorporate more from here
- No one expects you to know the most efficient keystrokes when you are starting out
- Good Luck on becoming a Vim Guru :)

Special Thanks

- Daniel Lindsay-Shad



Daniel
dan1e1z

Q & A