

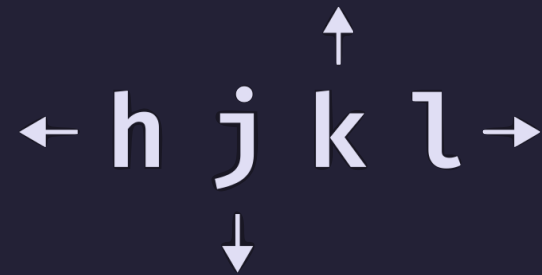
# Vim Editor



Frank Würthner

# Basic Navigation and Editing

Vim **Modes**: Normal, Insert, Visual, Command Line (Press <Esc> to return to normal mode)



Exiting Vim: :q, :w, :wq, :q!, :x

Start writing by pressing i or a (insert left or right of cursor)

Type :help to open up the help menu.

Think **operators**, **text objects**, and **motions**

# Operators

c	Change
d	Delete
y	Yank into register
~	Swap case
gu	Make lowercase
gU	Make uppercase
!	Filter to external program
<	Outdent
>	Indent
=	Indent

# Text Objects

Text objects allow us to interact with plaintext, parentheses, quotes, tags, and other common patterns that appear in text.

## Normal Mode

`<number><command><text object or motion>`

- `iw`: Inner word (word without surrounding whitespace)
- `aw`: Around word (word with surrounding whitespace)
- `is`: Inner sentence
- `as`: Around sentence
- `ip`: Inner paragraph
- `ap`: Around paragraph
- `it`: Inner tag
- `at`: Around tag
- `i(, i), i[, i], i{, i}`: Inside parentheses, brackets, or braces
- `a(, a), a[, a], a{, a}`: Around parentheses, brackets, or braces

# Delete, Put, Yank

## Normal Mode

- x: cut one character
- dd: cut one line
- daw: cut current word including spaces
- dfx: cut including first occurrence of char x
- yy: copy line
- y\$: copy to the end of line
- yiw: copy current word without spaces
- ytx: copy before first occurrence of char x
- p: paste after
- P: paste before
- gp: paste after and move cursor after pasted text

## Command Mode

- :%y: Yank all lines
- :1,20 co.: Copy from line 1 to 20 and paste it to cursor position

## Undo/Redo

- u: Undo last change
- Ctrl+r: Redo last undone change







# Advanced Navigation

## Line Navigation

`0`: Beginning of the line

`^`: First non blank character

`gg`: first line on the file

`g_`: last non blank character

`$`: End of the line

`G`: Last inline of the file

`g;`: go to last edit

`{linenumber}G`: go to specific line

`%`: Jump to matching bracket/parenthesis

`f{char}`: Jump to next occurrence of {char}

`t{char}`: Jump before next occurrence of {char}

## Line Scroll

`Ctrl+e`: Scroll one line down

`Ctrl+y`: Scroll one line up

## Viewport Scroll

`Ctrl+d`: Scroll half page down

`Ctrl+u`: Scroll half page up

`Ctrl+f`: Scroll one page down

`Ctrl+b`: Scroll one page up

## Word Navigation

`w`: Beginning of the next word

`e`: To the end of next word

`b`: Beginning of the previous word

`ge`: To the end of previous word

# Substitute

The substitute command allows us to find and replace text.

## Command Mode

`:s/javascript/python/g`

`:%s/css/sass/g`

`:5,12s/dev/production/g`

`:. ,+2s/bug/feature/g`

`:g/^review/s/needs/done/g`

`:%s/chrome/firefox/gc`

`:%s/text/Editor/gi`

`:%s/\<static\>/dynamic/g`

`:%s/remember/me/&`

`:%s/remember/me/n`

`g`: global (each occurrence changed, not the first)

`%`: substitute in all lines

`^review`: only in lines that start with review

`5,12`: only these lines

`. ,+2`: current and the next 2 lines

`c`: ask for confirmation

`i`: ignore cases

`&`: reuse last arguments

`\<...\>`: only whole words matching

## Normal Mode

- `s`: substitute a character
- `S`: substitute a line
- `r`: replace a character
- `R`: Replace over characters



# Vim Windows

A window is a viewport onto a file (buffer). You can use multiple windows on one buffer, or several windows on different buffers.

- `:sp`: Split the window horizontally
- `:vs`: Split the window vertically
- `:sp filename`
- `:q`: Close the current window
- `Ctrl+w w`: Switch between windows
- `Ctrl+w h/j/k/l`: Move to the window on the left/down/up/right
- `Ctrl+w +/-`: Increase/decrease the height of the current window
- `Ctrl+w </>`: Increase/decrease the width of the current window
- `Ctrl+w =`: Resize windows to be of equal size

# Plugins and Configuration

Plugin Manager: Vundle, Pathogen, vim-plug, lazy-vim

## Configuration

- `set number`: Display line numbers
- `set relativenumber`: Display relative line numbers
- `syntax on`: Enable syntax highlighting
- `set autoindent`: Enable auto-indentation
- `set tabstop=4`: Set tab width to 4 spaces
- `set shiftwidth=4`: Set indentation width to 4 spaces
- `set expandtab`: Convert tabs to spaces

## Popular Plugins

- NERDTree: File explorer sidebar
- ctrlp.vim: Fuzzy file finder
- vim-surround: Easily surround text with brackets, quotes, tags, etc.
- vim-airline: Enhanced status bar
- vim-gitgutter: Show git diff in the sign column

# Tips and Tricks

: '<, '>norm I-

xp

.: Repeat the last change

Ctrl+i: Jump forward to the next location

Ctrl+o: Jump back to the previous location

Ctrl+r: Insert the contents of a register

Ctrl+n: Auto-complete words based on the current file

Ctrl+p: Auto-complete words (reverse direction)

Macros