

Vim for Everyone

Linuxtage 2022

Christoph Lipautz

April 22, 2022

Presentation Setup

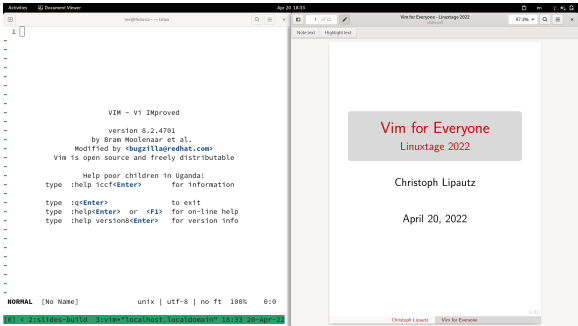


Figure 1: Screenshot: Left Panel Vim, Right Panel Slides

Overview

Vim is a powerful text editor, we will learn to use it to:

- Edit Text Efficiently
- Automate Simple Text Manipulation

My History with Vim

Long story short:

I switched from Kate to Vim
because I lost a bet.

Context Switch



Figure 2: Woodworking Workbench¹

¹<https://unsplash.com/photos/HjFUevA2g1k>



Figure 3: Have proper Tools²

²https://unsplash.com/photos/fqsT_W5JwSc



Figure 4: Have a Plan³

³<https://unsplash.com/photos/dQf7RZhMOJU>



Figure 5: Go for it⁴

⁴https://unsplash.com/photos/maxl1X_RJBY

First Step to Understand Vim

Vim works in modes!

- INSERT Mode ... write text
- NORMAL Mode ... manage text, navigate around
- COMMAND Mode ... run a command (e.g. save file, exit, search & replace, etc.)
- Others (VISUAL Mode, etc)

NORMAL, press `i`

INSERT, type stuff, exit `<ESC>`

NORMAL, press `:`

COMMAND, type `write<ENTER>`

NORMAL, press `v`

VISUAL, exit `<ESC>`

Quit Vim

If you don't know anything about **modes**, Vim is just a random password generator.

```
:help quit
```

...

```
:q[uit]! ⇒ quit and don't warn  
about unsaved changes
```

Mnemonics: Think of...

NORMAL mode actions are just keystrokes.

We want to **i**nsert some text.

We want to **v**isualize some text.

`w` ... move one `w`ord

NORMAL

More Navigation

w ... move one **w**ord

e ... move to the **e**nd of a word

b ... move **b**ack a word

f ... **f**ind character (in current line)

NORMAL

Navigation At Your Fingertips

Not every action has mnemonics

- `x` ... **cross out**, delete character under cursor
- `0`, `^`, `$` ... **beginning**, **start** (of text) and **end** of line
- Use `h`, `j`, `k`, `l` instead of arrow keys (optional but strongly recommended)

NORMAL



Figure 6: Vim Arrow Key Notes on Keyboard

Delete

`dw`

NORMAL

Delete a word.

de

db

NORMAL

Delete some words.

Odfa

Jump to the beginning of a line and delete until the first **a** character.

Golf Score: 4

Vim Golf

The screenshot shows the VimGolf website interface. At the top, there's a browser tab titled 'VimGolf - réal Vim ninjas' and the URL 'https://www.vimgolf.com/challenges/5fa95fbd285680008e41e4b'. The page features the VimGolf logo and a 'SIGN IN WITH TWITTER' button. The main heading is 'Real Vim ninjas count **every** keystroke - do you?'. Below this, it says 'Pick a challenge, fire up Vim, and show us what you got.' and provides links for 'Changelog', 'Rules & FAQ', and 'updates: @vimgolf, RSS.'.

The challenge is titled 'swap number pairs' with the description 'Swap the numbers in a bunch of 2-element arrays.' The 'Start file' contains the following content:

```
[2, 1], [5, 4]
[6, 3]
[7, 4]
[8, 2], [12, 11]
```

The 'End file' contains the following content:

```
[1, 2], [4, 5]
[3, 6]
[4, 7]
[2, 8], [11, 12]
```

There are links for 'View Diff' and 'Solutions'. The 'Solutions' section includes a note: 'The best way to learn is to practice. Below, you will find some of the solutions other golfers have entered. To unlock higher ranked solutions, submit your own entry which does as well or better than the solutions you can currently see - climb the ladder!'.

On the right side, there's a terminal window showing the VimGolf key and installation instructions:

```
Your VimGolf key: please sign in
$ gen install vimgolf
$ vimgolf setup
$ vimgolf put 5fa95fbd285680008e41e4b
```

Below the terminal, it says 'Created by: @wedestoft' and '272 active golfers, 1374 entries'. A 'Leaderboard (lowest score wins):' section lists the top 5 players:

Rank	Player	Score
#1	Hugh / @haquaman	16
#2	hepebexe / @hepebexe	16
#3	John Braxder / @braxder	16
#4	Alice / @Alice92541388	16
#5	Andreas Näsman / @andreas_nasman	16

Figure 7: Screenshot of www.vimgolf.com

Delete a Word

We can combine commands:

- `dw` ... `d` delete a `w` word
- `de`, `db`
- `df<char>`
- `d$` or shorthand `D`
- `dd` shorthand to delete a line

Change a Word

`cw` ... delete a word and switch to insert mode

`ce`, `cb`, `cf<char>`, `c$` or `C`, `cc`

Visualize a ...

```
fwveeedx
```

Visualize a word word word word
then delete them.

The . Command

The `.` command is a superfast way to repeat the last action.

```
wwcethree<ESC>jjb.
```

1 2 3 4 5

one two two four five

one two two four five

Manual Search & Replace

- `/` ... search
- `cw` ... `c`hange `w`ord
- `n` ... `n`ext found text
- `.` ... repeat action (e.g. `cw`, `d3w`, ...)

; Repeat Find

`f<char>;` ... repeat the `f`ind

`fa;x`

Move around conveniently with find
aand repeat.

The Macro Command

Start recording:

`q<char>`, e.g. `qq`
for using register `q`.

Repeat recorded automation:

`@<char>`, e.g. `@q`.

Vim Automation

Can be helpful if we have a one-time job at hand and search & replace won't do the job.

For automation of regular/recurring jobs please stick to scripting.

Hint: Visual Cues

`cip` changes the current paragraph (under cursor), that can lead to an unexpected result.

Instead use `vip`, then decide if the expected selection is given and press `c` to change it.

Hint: Never Stop Learning

- Master switching to insert using `i`, `a`, `I`, `A`, `o`, `O`.
- Figure out the possibilities with command mode actions.
- Checkout Plugins (fuzzy finding, file opening, file browsers, code formatter, testsuite runner, etc).

Examples

- Swap un/ordered Lists (Markdown)
- Manipulate CSV Files
- Edit your Shell Commands

Resources

- Run `$ vimtutor`
- Find some Vim Community
- Your very welcome to research, use or discuss my Vim configuration at github.com/unused/.vim

The screenshot shows the GitHub interface for the repository 'unused/vim-for-everyone'. The repository is public and has 8 commits. The file list includes 'assets', 'examples', 'templates', '.gitignore', 'LICENSE', 'Makefile', 'README.md', and 'slides.md'. The 'README.md' file is selected, showing the title 'Vim For Everyone' and a description: 'A presentation for Vim Editor. It introduces Vim modes, shows basic navigation controls, how to combine actions in normal mode and explains some basic automation/repetition tools.' It also includes a link to 'my vim configuration' and a section titled 'Summary of Commands Used' with a list of commands: 'h, j, k, l ... left, down, up, right, use instead of arrow keys'.

Search or jump to... Pull requests Issues Marketplace Explore

unused / vim-for-everyone Public

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags Go to file Add file Code +

unused Extend general documentation 8d4a3c 2 days ago 8 commits

assets	Add more photos to slides	2 days ago
examples	Int presentation	4 days ago
templates	Int presentation	4 days ago
.gitignore	Int presentation	4 days ago
LICENSE	Initial commit	4 days ago
Makefile	Int presentation	4 days ago
README.md	Extend general documentation	2 days ago
slides.md	Add more photos to slides	2 days ago

README.md

Vim For Everyone

A presentation for Vim Editor. It introduces Vim modes, shows basic navigation controls, how to combine actions in normal mode and explains some basic automation/repetition tools.

Have a look at [my vim configuration](#) in case you wanna figure out configuration and plugins I recommend.

Summary of Commands Used

- `h, j, k, l` ... left, down, up, right, use instead of arrow keys

Figure 8:
github.com/unused/vim-for-everyone