

# (neo)vim

Paul Schaefer

11. Mai 2018



1. Introduction

2. Basic Usage

3. Useful Plugins

4. Next Steps

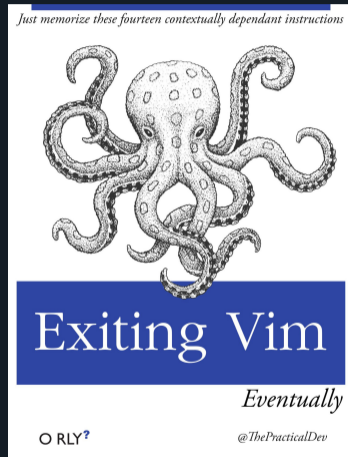


## Out of Scope:

- vim vs. <editor of choice>
- Not even vim vs. emacs
- Not even vim vs. neovim

## In Scope:

- Usage and configuration of **vanilla** vim
- Being **productive**
- Some useful **plugins**
- Just **my favorite** plugins. There are others.





Vim is **different**.

Vim has **modes**.

1. Normal Mode
2. Insert Mode
3. Command Mode
4. Visual Mode

Vim does **not** use **arrow keys**.

- **h** moves left
- **j** moves down
- **k** moves up
- **l** moves right
- Several other movements are available



Just tell vim *where* you want your *cursor* to be.

„Three lines below“	3j
„At the next word“	w
„At the end of this word“	e
„Find the next paren“	f)
„Find the previous paren“	F)
„In line 17“	17G
„In the first line“	gg
„In the last line“	G



Speak to vim in **sentences**:

delete

three

words



Speak to vim in **sentences**:

d

three

words



Speak to vim in **sentences**:

d

3

words





Speak to vim in **sentences**:

d

3

W



Speak to vim in **sentences**:

**Verb**

d

**[Quantifier]**

3

**Object**

W



Speak to vim in **sentences**:

## Verb

d  
c hange  
y ank (copy)  
v isually select  
> increase indent  
= fix indent  
...

## [Quantifier]

3

## Object

w  
p aragraph  
h left  
j next line  
k prev line  
l right  
i} in }'s  
...

## Inserting text



i Insert text at cursor

Test line

a Append text after cursor

Test line

I Prepend text at the beginning of the line

Test line

A Append text at the end of the line

Test line|



# DEMOTIME



As there is no menu, there are **commands**, entered by `:command`.

- Basic commands
  - `:w`, `:write`
  - `:e`, `:edit`
  - `:tabe`, `:tabedit`
- Setting options with `:set`
  - `number`, `relativenumber`
  - `syntax`
  - `shiftwidth`, `tabstop`, `expandtab`
  - `autoindent`
  - ...



Vim is similar to [sed](#).

**Search:**        /<sup>^</sup>Full line\$  
                 /<sup>^</sup>Beginning of line  
                 /End of line\$  
                 /anywhere

**Relevant options**  
smartcase/nosmartcase,  
ignorecase/noignorecase

**Replace:**        :%s/<sup>^</sup>Full Line\$/replace with.../g  
                  :%s/<sup>^</sup>Full Line\$/replace with.../gc  
                  %s/Number \([0-9]\+\) is weird./I like number \1 best!/g



Vim can work with multiple files in **splits** and **tabs**.

- **:tabe** <filename> opens the file in a new tab
  - **gt** and **gT** iterate through tabs
  - **:q** closes the current tab and exists if last tab
  - **:qall** closes all tabs and exists vim
- **:vs** <filename> and **:sp** <filename> opens a file in a split
  - Splits are *inside* tabs
  - **:e** <filename> opens another file in the active view

```
2 ~/.vim/init.vim  ~/.main.c
1
2
3 " key mappings
4
5 nnooremap ; :
6 nnooremap <F4> :set invhsearch<CR>
7 innooremap <F4> <ESC>:set invhsearch<CR>gi
8
9 vmap <C-d> :EasyAlign -/ \*!~/<10>r0<CR>
10
11 ~/.vim/mappings.vim 1,1 Top
12
13 filetype plugin indent on
14 syntax on
15
16 [ ] set number relativenumber
17
18 1 set mouse=a
19 2 set showcmd
20 3 set autoindent expandtab tabstop=4 shiftwidth=
21 4 set clipboard=unnamedplus,unnamed
22 5
23 ~/.vimdemo/init.vim 15,1 Bot
```



## Minimal configuration



```

call plug#begin('~/.vim/plugged')
  Plug 'mhartington/oceanic-next'           " colorscheme
  Plug 'Shougo/deoplete.nvim',             { 'do': ':UpdateRemotePlugins' } " Input Completion
  let g:deoplete#enable_at_startup = 1
  Plug 'machakann/vim-highlightedyank'     " Visible copying
  let g:highlightedyank_highlight_duration = 500
call plug#end()

colorscheme OceanicNext                   " Use colorscheme

filetype plugin indent on              " Load filetype specific configurations
syntax on                                " Enable syntax highlighting

set number relativenumber                " Show line numbers and relative ones
set mouse=a                              " Enable mouse integration
set showcmd                              " show the commands inserted
set autoindent expandtab tabstop=4 shiftwidth=4 " Autoindent with 4 spaces
set clipboard=unnamedplus,unnamed       " Use system clipboard

```

### Top Plugins (that didn't fit on the previous slide)



You should definitely use a [plugin manager](#).

- <https://github.com/junegunn/vim-plug>
- `Plug 'mbbill/undotree'`
- `Plug 'ctrlpvim/ctrlp.vim'`
- `Plug 'SirVer/ultisnips'`
- `Plug 'honza/vim-snippets'`
- `Plug 'aperezdc/vim-template'`
- `Plug 'vim-airline/vim-airline'`



# DEMOTIME

... What about awk, grep and the other unix tools?!



... What about awk, grep and the other unix tools?!



They are **not obsolete!**

- `!` calls an external program
  - selected text is given as stdin
  - selected text is replaced by stdout
- `:r` reads an external file
- Example:

```
:r /etc/passwd  
vip  
!awk -F: '{if ($3 > 50) { print $1; } }'
```



- Undofiles
- `:set incsearch`
- Colorschemes
- Spell Checking with `:set spell`
- More Plugins!!!!1111
  - `Plug 'tpope/vim-surround'`
  - `Plug 'tpope/vim-repeat'`
  - `Plug 'tomtom/tcomment_vim'`
  - `Plug 'majutsushi/tagbar'`
  - `Plug 'scrooloose/nerdtree'`
  - `Plug 'mhinz/vim-grepper'`
  - `Plug 'atweiden/vim-dragvisuals'`
  - ...



- `:help`
- Google Is Your Best Friend
- Force yourself to use vim
- Look for other interesting Plugins
- <http://vimcolors.com/>



# Q & A



# (neo)vim

Paul Schaefer

11. Mai 2018