

## About

This is a collection of config files (dot files) for different tools.

Find bellow a small cheet shet for using `vim`, `neo vim` and `tmux`. Also this repository contains configuration files for mentioned tools.

[!warning] Install required packages via the `install/install_brew.sh` first

---

## Legacy documentation

In order some Plugins are able to work, `pip3` and `pynvim` should be intalled via

```
sudo pip3 install --upgrade pynvim
```

How to install `pip3` see instructions for appropriate OS.

## Tool for installing plugins for ZSH

`zplug` is used for managing `zsh` plugins.

To install `zplug`:

```
brew install zplug
```

## Vim and NeoVim plugin is used

Used the following vim plugin: <https://github.com/junegunn/vim-plug>

## Neo vim

### Aliases for Neovim

Add to `.bashrc` or `.zshrc`

```
alias vim="nvim"  
alias vi="nvim"
```

Nvim uses config file at: `~/.config/nvim/init.vim`

`vim-plug` should be additionaly isntalled. See details in README  
<https://github.com/junegunn/vim-plug>

## Vim hot keys

```
h - move left  
l - move right
```

```
j - move down  
k - move up
```

## Insert Mode

a - move cursor at 1 symbol right for editing  
A - move cursor at the end of line for editing  
o - move cursor at the new line for editing  
O - create a new line above for adding

## Copy-Paste

yy or Y - copy line  
dd - delete (cutting) the line  
  
p - paste copied(deleted) text after current line  
P - paste copied(deleted) text before current line  
yw - copies a word under cursor

## Navigation

gg - move cursor at the beginning of the file  
G - move cursor at the end of the file  
w - move to the next word  
b - move to the word back (backward)  
e - move to the end of the next word  
dd - delete current line  
% - move to the pair symbol (braces, etc)  
  
`. - move cursor to the last edit position  
` + symbol - go to saved bookmark, where symbol - the name (single character) for bookmark  
`D - go to the global bookmark`  
gf - go to the file under current cursor (e.g. new File(/my-file.txt) will go to the my-file.txt)  
Ctrl + g - shows your location in the file and the file status.  
G - move cursor to the end of file  
gg - move cursor to the beginning of the file  
20gg - will move cursor to specified line number 20  
  
Ctrl + O - returns back to the place where cursor was before search  
j\$ - move cursor to the end of the next line

## Repeat command

Into command mode print 2w to move to the next word twice

## Searching

f + symbol - find specified `symbol` at current line (; finds the next matching symbol)  
F + symbol - find specified `symbol` back  
/ + any symbol or word to find it at the whole document from top to bottom  
n - find next matching search result (used with /). This uses word Case.  
N - navigate Up to the next found word  
? + any symbol to find `symbol` at the whole document from bottom to top.

:set ic - option for ignoring case during the search  
:set hls is - highlights found words

t + symbol\_to\_be\_find go to position before searching symbol, whe symbol\_to\_be\_find is a sym  
T + symbol - find the specified symbol and move cursor befor it backward  
\* - find thw word unser cursor (all found words in the file will be selected)  
\ - ignore a symbol during the search  
noh - turn the highlight off for the found word  
dt + symbol - delete everything befor specified symbol  
df + symbol - delete everything before the specified symbol  
d\$ or shift + d - deletes from the current cursor position to the end of the current line

## Additional notes

NOTE: To remove the highlighting of matches enter: :nohlsearch NOTE: If  
you want to ignore case for just one search command, use in the phrase:  
/ignore<ENTER>

Some usefull set commands:

ic - 'ignorecase'	ignore upper/lower case when searching
is - 'incsearch'	show partial matches for a search phrase
hls - 'hlsearch'	highlight all matching phrases

## Replace

s/old/new - replaces first matching of `old` text with `new` value into current line  
s/old/new/g - replaces all matches of `old` text with `new` value into current line  
#,#s/old/new/g - changes every occurrence of a character string between two lines, where #,#  
%s/old/new/g - changes every occurrence in the whole file  
:%s/old/new/gc - to find every occurrence in the whole file, with a prompt whether to subst

## Edit

r + symbol - replace char under cursor by specified symbol  
ce - change text from cursor to the end of the word  
c\$ - change text from cursor to the end of line.

## Find into project's files

This can be achieved either by `vimgrep` or `grep`

```
vimgrep def * - find any matches of def at all files
cnext - go to the next found matches
cprev - go to prev found matches
```

`grep` found all matches and display found results into terminal without vim integration.

## Bookmarks

`m + bookmark_name` - create a bookmark with the name ``bookmark_name`` for current line.

`mD - symbol` - create a global bookmark with the name ``symbol`` (instead of D can be another c

## Windows

With user-defined function:

function looks like

```
function! WinMove(key)
    let t:curwin = winnr()
    exec "wincmd ".a:key
    if(t:curwin == winnr())<LeftMouse>
        if (match(a:key, '[jk]'))
            wincmd v
        else
            wincmd s
        endif
    endif
    exec "wincmd ".a:key
endfunction
```

Using hotkeys

```
Ctrl + k opens a new window above current or switch to window above
Ctrl + h opens a new window at right or switch to the right window
Ctrl + l opens a new window at left or switch to left window
Ctrl + j opens a new window at bottom or switch to the bottom window
```

## Tmux

tmux will run a new tmux

```
Ctrl + B + % - split screen vertically
Ctrl + B + " - split screen horizontally
Ctrl + B + arrows - move between tmux screens
Ctrl + B + o - swap panes
```

Ctrl + B + q - show panes numbers  
Ctrl + B + x - kill pane  
Ctrl + B + space - toggle between layouts

## Resize tmux windows

Ctrl + B:resize-pane -L 20 - decrease current window to 20 to left

Ctrl + B:source-file ~/.tmux.conf - reload tmux config file

Ctrl + B + n - create a new tmux window

Ctrl + B + 1 - move to the 1st window

Ctrl + B + \$ - rename current tmux's session

tmux attach -t session\_name - attach to specified session with name `session\_name`

## Additional useful links

Platform IO for Vim