Introduction to vi

how to use neovim

장창서 <changseo.jang@riiid.co>

Riiid

December 22, 2022

개요

역사

키정의

모드

기본적인 움직임

기본적인 커맨드 구조

몇몇 유용한 커맨드

설정파일

플러그인 매니저

역사

ed EDitor

vi VIsual

vim VI iMproved

neovim vim 리팩토링 버전

- vimscript 구버전은 호환가능
- lua
- 내장 lsp client가 있음

키정의

C **C**trl

M Alt / Option / **M**eta

CR Enter / Return (**C**arriage **R**eturn)

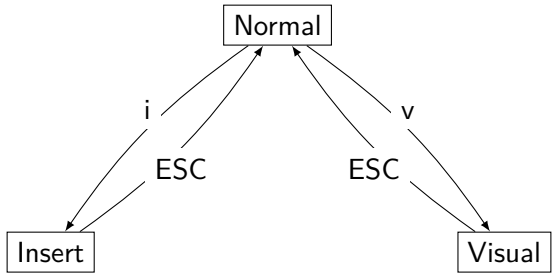
모드

Normal 커맨드들을 사용할 수 있는 모드

Command(Ex) Normal 모드에서 colon을 입력하면 진입하는 실행 모드

Insert 글자들을 넣을 수 있는 모드 (보통 에디터처럼 사용할 수 있음)

Visual 에디터에서 마우스로 드래그하는 것과 비슷한 선택을 보여주는 모드



모드

Visual

v	visual 모드, 문자 하나씩 선택할 수 있음
C-v	v-block 모드, 블럭으로 선택할 수 있음
V	v-line 모드, 라인으로 선택할 수 있음

기본적인 움직임

화살표 h(←) j(↓) k(↑) l(→)

문자 i(앞에 insert) a(뒤에 insert)

단어 b(**b**ack) w(**w**ord) e(**e**nd)

문장
- I(문장 맨 앞에 insert) A(문장 맨 뒤에 insert)
- o(**o**pen, 문장 아래에 insert) O(문장 위에 insert)
- ∧(문장 맨 앞으로) $(문장 맨 뒤로) ⇒ regex

문단 {(이전 문단) }(다음 문단)

기본적인 커맨드 구조

(count +) action + (modifier +) range

- **count** 반복횟수
- **action**
  - y: **y**ank
  - d: **d**elete → 동시에 yank됨
  - c: **c**hange
  - v: **v**isual
- **modifier**
  - i: **i**nside
  - a: **a**rround
- **range** 움직임 + pair

기본적인 커맨드 구조

예시1 - 움직임

- `vw`: 단어 하나 선택
  - v visual
  - w word
- `dw`: 단어 하나 삭제
  - d delete
  - w word
- `cw`: 단어 하나 변경
  - c change
  - w word

기본적인 커맨드 구조

예시|2 - pair

- `vi}`: 중괄호 안쪽 선택
  - v visual
  - i inside
  - } 중괄호
- `va}`: 중괄호 포함 선택
  - v visual
  - a around
  - } 중괄호

몇몇 유용한 커맨드

dd	문장 삭제
gg	맨위로
G	맨아래로
C-o	**o**ut, 이전 위치로
C-i	**i**n, 다음 위치로

설정파일

~/.config/nvim/init.vim

```
let mapleader = ";"

runtime vim/others.vim
```

~/.config/nvim/vim/others.vim

```
imap <silent>,d <ESC>

set wildmenu
set ignorecase
set smartcase
set hlsearch
set incsearch

set number

set autoindent expandtab tabstop=2 shiftwidth=2
```

설정파일

lua

~/.config/nvim/init.vim

```
    ...

    lua require('boot')
```

~/.config/nvim/lua/boot.lua

```
    vim.api.nvim_command('echo "lua loaded"')
```

플러그인 매니저

vim-plug
- 속도가 빠름
- https://github.com/junegunn/vim-plug

packer.nvim https://github.com/wbthomason/packer.nvim

플러그인 매니저

설치

Shell

```
git clone --depth 1\
  https://github.com/wbthomason/packer.nvim\
  ~/.local/share/nvim/site/pack/packer/start/packer.nvim
```

~/.config/nvim/lua/boot.lua

```
require('plugins')
```

~/.config/nvim/lua/plugins.lua

```
vim.cmd [[packadd packer.nvim]]

return require('packer').startup(function(use)
  use 'wbthomason/packer.nvim'
end)
```

플러그인 매니저

패키지 추가하기

~/.config/nvim/lua/plugins.lua

```lua
return require('packer').startup(function(use)
  use 'wbthomason/packer.nvim'

  use 'username/reponame'
end)
```

install :PackerInstall

sync :PackerSync

clean :PackerClean

플러그인 매니저

패키지 추가하기

chadtree https://github.com/ms-jpq/chadtree

~/.config/nvim/lua/plugins.lua

```
use {
  'ms-jpq/chadtree',
  branch = 'chad',
  run = 'python3 -m chadtree deps',
}
```

~/.config/nvim/vim/others.vim

```
...

nmap <silent><leader>e :CHADopen<CR>
```

플러그인 매니저

패키지 추가하기

tree-sitter https://github.com/nvim-treesitter/nvim-treesitter

~/.config/nvim/lua/boot.lua

```
...

require('others')
```

~/.config/nvim/lua/others.lua

```
require('nvim-treesitter.configs').setup {
  highlight = { enable = true },
}
```

플러그인 매니저

패키지 추가하기

nvim-lspconfig `https://github.com/neovim/nvim-lspconfig`

플러그인 매니저

패키지 추가하기

이외에 유용한 플러그인들

- `https://github.com/nvim-telescope/telescope.nvim`
- `https://github.com/numToStr/Comment.nvim`
- `https://github.com/lukas-reineke/indent-blankline.nvim`
- `https://github.com/airblade/vim-gitgutter`

EOF