Vim commands

Andy Kim

Basic movements

- $ touch ~/.vimrc or :set nocompatible : enables VIM functionalities
- gvim vs vim: gvim opens a new window
- <insert mode> ctrl+w, ctrl+h delete a word, a char
- 3a!<esc> inserts !!!
- help i_ shows <insert mode> commands
- help i_ctrl+h shows help of insert mode ctrl+h
- help . . . : refer to the manual
- inserting multiple lines at VIM: <visual mode> ctrl+v, choose multiple lines, shift+i, edit, <esc>
- 3G, 10G, nG: go to nth line
- :set number, :set nonumber
- ctrl+G: shows where I am
- 30%, n%: go to n% position of the buffer
- ctrl+U vs ctrl+D: up vs down
- ctrl+E vs ctrl+Y: up vs down
- d3w vs 3dw: identical
- C vs S: Clear the rest of the line and insert mode, delete all line and insert mode
- .: repeat the last change
- df<char>: delete until <char> in a line
- J: joining the next line, also works within a visual mode
- r: replace a char
- ~, 3~, 10~: change cases
- Macros:
    - Recording: q[a-z]
    - Ending: q
    - Using: @[a-z]
- Digraphs:
    - ctrl+k+<key> in <insert mode>

- :diagraph
- Search:
  - / ?: forward/backward search
  - use escape(\) for special characters (. * [ ] ^ % ? ~ $)
  - n N: forward/backward navi
  - / and up-down arrow for history search
  - ^text text$ tex.: start end replace with a char
- :set hlsearch / :set nohlsearch / :nohlsearch (to clear current highlights)
- :set incsearch / :set noincsearch
- Marks:
  - Setting: m[a-z]
  - Using: '[a-z] (backtick to the mark), '[a-z] (single quotation to start of the line)
  - :marks
  - ": last place after jump
- Yanking: y'a could yank all the lines up to mark a
- :vi <file> editting another file
- Register yanking / <insert mode> pasting
  - <visual mode> "[number]y to yank
  - <insert mode> ctrl+r [number] to paste
- 0: moves to the start of a line
- ctrl+[is the same as <Esc>

Advanced

- Window split:
  - :split duplicates current working file
  - :new opens a new empty vim
  - :buffers shows buffers
  - :buffer number/filename shows buffer
  - ctrl+w, j k: moves about the panes
- Multiple file editing
  - $vim file1 file2 file3
  - :next or :n or :ne or :3n moves windows
  - :prev or :3prev moves to previous windows
  - :args shows windows
- Visual mode multi line editing:
  - > < command indents dedents "shift width"

- – ctrl+v to enter <visual model>, and then shift+i (or I), then type text, and <Esc>
  - – also works with A (append), C (delete from block to the end of line)
- >> / << also indents / dedents a line
- 5>> / 5<< applies indents / dedents for the next 5 lines
- syntax on: for syntax coloring
- Example
  - – :filetype on
  - – :autocmd FileType c,cpp :set cindent
- Abbreviation:
  - – :abbreviate ad advertisement (:ab also works)
  - – :ab shows the list
- Mapping (similar to macro)
  - – :map <F5> i{<Esc>ea}<Esc>
  - – :map shows the list
- Saving and restoring setting (e.g., ab or mapping)
  - – :mkvimrc <file>
  - – :source <file>
- :version shows which initialization file used
- :set textwidth=30
- :set fileformat=unix,dos / :set fileformat? shows the list

Customized

- :inoremap <C-j> <Esc> defined in .vimrc enables <C-j> to delete current char (i.e., delete) in <Insert Mode>