



# Vim ?????

????

???????????????	????	????
i	???????????????	? a ??????????????????????????????
a	???????????????	
o	?????????????	
h j k l	?????????h--??, j--??, k--??, l--??	?????????????????????????????????
:1 : 22	????????? ??????22?	???????????????????
gg	???????	
G	???????	?????????????????????????????
O / \$	??????/??	
w / W	??????????/??	?????
b / B	??????????/??	?????
x	???????	
J (Shift + j)	???????????	?????????
{	???????????	
}	???????????	
:wq	???????	

:w new.txt	????? new.txt	
:w !sudo tee %	? sudo ????	???????????????? sudo ???????
:q!	?????	?? ?
:set nu!	??????	
:set hlsearch	Highlight ??????	???????????
:set cuc!	??????	?? json, yaml ??
: syntax on	?? Hihjlight ??	
d dd	???????? ????????	
10dd	???????10?(???)	
y yy	???????? ????????	
3yy	???????3?(???)	
p	??	
/keyword	????	
^KeyWord	????(?????)	
/S[Uu]SE	????????????	
*	????????????	
:%s/old/new/g :%s/old/new/gc	????????old????????new?? gc: ????	
:s#/usr/local/bin#/usr/sbin#g	? # ??????	
:10,30 s/old/new/g	?10-30??? old ? new	

:%s/old1\ OLD2/new/gc	???? old1?OLD2 ? new	
:/<table>/,/</table>/g/^\$/d	?? <table> ... </table> ?????	
:%s\t/ /g	?? Tab ?4???	???????
^\s+\$	???????????? Highlight	????? :set hlsearch ??, not support AIX
:%s\s\+\$//	??????????????	not support AIX
~	??????	
:%s/[ctrl+V and ctrl+M]/g	?????? ^M	
ggVG	????	?? Ctrl a
dG	????????????	
dgg	????????????	
:1,\$d	??????	
:1,22d	?? 1-22 ?	
:23,\$d	?? 23 ????????	
:! ls -l	???? SHELL ??	!! ??????
:vsplit my.file	???????????? my.file	ctrl + w + w ????
:split my.file	??????	ctrl + w + w ????
:sort :sort! :sort u :sort i :sort n	?? ???? ????????? ????????? ??????	
u Ctrl + r	Undo Redo	

:g/^\$/d :4,100g/^\$/d :g/pattern/d :!g/pattern/d	?????? ?4-100????? ???? pattern ?? ????? pattern ??	[range]g/pattern/cmd
--	--	----------------------

?????

- ?????????? '.'
- ???????? :terminal
  - :below terminal , :bel term
  - :vertical terminal , :vert term
  - ???? : Ctrl + w + w
  - ???? : Ctrl + w + N ??????: hi ?
- ???????? :find your-script.sh ??? tab ????? \* ???
  - :find \*.sh

## Visual mode

- ??????: V
- ??????: Shift + V
  - ???? : Shift + V
  - ??????????: V
- ??????: Ctrl + V
  - ??????????: I
  - ??????????: A
  - ??????????????: \$
- ??????
  - ?? : ?/??
  - ??????: w
  - ??????????: f .
  - ??????: }
- ??????: Esc
- ??????????: y
- ??????: p
- ??????????: d \*(Delete)\*
- ??????????: d > Move to target line > p
- ??????????: r \*(Replace)\*
- ??????????: c \*(Change)\*
- ??????????: u \*(Undo)\*
- ??????????("?")?????: i + " ? '

?????????

- ?? MyKeyWord? /MyKeyWord
- ????????????????? \*?

- ??????:set hlsearch
- ??????:set incsearch
- ?? { }, [ ] ( ) ?????? %
- ?????????????? { ? [ {
- ?????????????????? zt (Top), zz (Middle), zb (Bottom)
- ?????????????? H (Top), M (Middle), L (Last Line)
- ?????? Ctrl+b (???), Ctrl+f (???)
- ?????? Ctrl+y (??), Ctrl+e (??)

## ??????

- ?????????? /old -> cw -> ?????? new -> ?????? Esc -> ?????? n -> ?????? .
- ?????????
  1. ???shift + v
  2. ???
  3. ?????? :s/old/new/g (?????????????)

## ????

```
# ????????????????????????????????????????? p
:<v>yank

# ????????????????????????????????????????? . copy t
:<v>copy.
:<v>t.

# ????????????? 10 ?????????????????????????
:+10t.

# ????????????? 10 ?????????????????????????
:-10t.

# ?????????????(33 ~ 44)????????????
:33,44t.
```

## ????????????????????????????????

1. Ctrl + v
2. ?????????????
3. ????????
4. Shift + I
5. ???????????
6. Esc ???????

?????????????`;

1. ??????????
2. Ctrl + **v**
3. ??????????
4. Shift + **\$**
5. Shift + **A**
6. ?????????? **;**
7. Esc ??????

????

1. Shift + **v**
2. ?????
3. **:norm i#** (???????)
4. **:norm x** (???????)

?????????

1. Shift + **v** (Visually Seleting)
2. ?????
3. **:w /path/to/newfile** (Save as newfile)
4. **gv** (reselect the last visual selection)
5. **d** (delete that selection)

?????? (auto-complete)

1. ??????
2. ?? /etc/profile??????
3. Ctrl + **x** -> Ctrl **f**

## Buffers

?????????

- **vim aa.txt bb.txt cc.txt** : ??????
- **:buffers** : ??????????????
- **:bn[ext]** : ??????????
- **:bp[revious]** : ??????????
- **:e filename** : ?????
- **:b filename** : ?????? (????????)
- **:b#** : ?????????

# Tutorials

- [Youtube Channel - Vim \(TheFrugalComputerGuy\)](#)
- [VIM Colors Online](#)
- [VimTricks](#)

# Vim FAQ

Err: A line cannot be longer than 2048 characters.

???? VI ??????????? 2048 ????????????????????????????????

```
cut -c1-2047 before.log > after.log
```

?? 3X MB ????????

“ ex: 0602-101 Out of memory saving lines for undo.

```
# the number must be less than 8 digits.
```

```
vi -y 99999999 your-file
```

???????

????????????????(autoindent)?????????????????

Solution: ?????????? `:set paste`

0602-140 The window is too small to display the current line.

AIX ????????????????????????

Solution:

1. ? putty ??????????????????? 9 ???
2. ?? Vim ?? Vi?

Unable to paste from clipboard on AIX

? Windows ?????????? putty ????????????

Solution: `:set mouse=v`



# vimrc

## vimrc simple

```
"
" Fixed for Python virtualenv as the follow error.
" ModuleNotFoundError: No module named 'powerline'
"
python3 import sys;sys.path.append("/home/along/.local/lib/python3.10/site-packages")
"
" Powerline with Vim
set laststatus=2
python3 from powerline.vim import setup as powerline_setup
python3 powerline_setup()
python3 del powerline_setup

"
syntax on          " Enable syntax highlighting
set number         " Show line numbers
set relativenumber " Show relative line numbers
set tabstop=4     " Set tab width to 4 spaces
set shiftwidth=4  " Set indentation width to 4 spaces
set expandtab      " Use spaces instead of tabs
set cursorline    " Highlight the current line
set paste         " Put Vim in Paste mode.
set ruler         " Show the line and column

" Moving Lines
" Ctrl-j: Move down, Ctrl-k: Move down
nnoremap <c-j> :m .+1<CR>==
nnoremap <c-k> :m .-2<CR>==
inoremap <c-j> <Esc>:m .+1<CR>==gi
inoremap <c-k> <Esc>:m .-2<CR>==gi
vnoremap <c-j> :m '>+1<CR>gv=gv
vnoremap <c-k> :m '<-2<CR>gv=gv
```

```
" Preserve last editing position
silent! source $VIMRUNTIME/defaults.vim

" Copy to Clipboard
" Prerequisites: apt install vim-gtk3
" Usage: Press v to select the lines, and then press y to copy
set clipboard=unnamedplus
```

## vimrc advanced

```
"
" minimal vimrc with no (extra) plugins
"

"regular settings
"-----

" ui
set number
set ruler
set wildmenu
set showcmd
set showmatch

" encoding/format
set encoding=utf-8
set fileformats=unix,dos,mac

" searching
set hlsearch
set incsearch
set ignorecase
set smartcase

" indent
set shiftwidth=4
set tabstop=4
set softtabstop=4
set autoindent

" allow syntax and filetype plugins
```

```

syntax enable
filetype plugin indent on
runtime macros/matchit.vim

" Moving Lines
" Ctrl-j: Move down, Ctrl-k: Move up
nnoremap <c-j> :m .+1<CR>==
nnoremap <c-k> :m .-2<CR>==
inoremap <c-j> <Esc>:m .+1<CR>==gi
inoremap <c-k> <Esc>:m .-2<CR>==gi
vnoremap <c-j> :m '>+1<CR>gv=gv
vnoremap <c-k> :m '<-2<CR>gv=gv

" autocmds
"-----
augroup general
  autocmd!
  "keep equal proportions when windows resized
  autocmd VimResized * wincmd =
  "save cursor position in a file
  autocmd BufReadPost * if line("\") > 1 && line("\")
    \ <= line("$") | exe "normal! g'\\" | endif
augroup END

augroup languages
  autocmd!
  autocmd BufNewFile,BufRead *.bash set syntax=sh
  autocmd FileType python xnoremap <leader>r <esc>:'<,'>:w !python3<CR>
  autocmd FileType go set noexpandtab
  autocmd FileType html :syntax sync fromstart
  autocmd FileType html,javascript,css,json,yaml,sh
    \ setlocal ts=2 sts=2 sw=2 expandtab
augroup END

```

## Learning

- [A Minimal Vimrc](#)
- [Configure Vim with vimrc](#)

# Cheat Sheets

PDF: [vim-commands-cheat-sheet-by-pnap.pdf](#)



## Vim Cheat Sheet



Created by [@dan\\_nanni](#) on Instagram



### Editing Mode

**:** command mode  
**ESC** normal mode  
**i** insert mode  
**v** visual mode, highlight characters  
**V** visual line mode, highlight lines

### Cursor Movement

**Character** **h** (left) **j** (down) **k** (up) **l** (right)  
**Word** **w** (next start) **b** (prev) **e** (next end)  
**Line** **0** (start) **\$** (end) **^** (first non-blank)  
**Screen** **Ctrl+f** (page down) **Ctrl+b** (page up)  
**File** **gg** (first line) **G** (last line)

### Editing Operation

**Insert** **i** (before cursor) **I** (line beginning)  
**a** (after cursor) **A** (line end)  
**o** (new line below) **O** (new line above)  
**Delete** **x** (current char) **dd** (or **:d**) (current line) **D** (to end of line) **dw** (word)

### Copy and Paste

**Copy** **yy** (or **Y**) (current line) **3yy** (3 lines)  
**yiw** (current word without whitespace)  
**yaw** (current word with whitespace)  
**y\$** (to end of line)  
**Paste** **p** (paste after) **P** (paste before)

### Undo and Redo

**u** Undo the last action  
**U** Undo all latest changes made to the line  
**Ctrl+r** Redo the last action

### Search and Replace

**Search** **/pattern** (from current cursor)  
**n** (next match) **N** (prev match)  
**Replace** **:s/old/new/g** (replace in cur line)  
**:%s/old/new/g** (replace in all lines)  
**:%s/old/new/gc** (replace with confirm)

### File Operation

**:w** Save current changes  
**:w <filename>** Save changes to another file  
**:q** Exit but it will fail with unsaved change  
**:q!** Exit without saving changes  
**:wq** (or **:x**) Save changes and exit  
**:e <filename>** Open a new file

### Window Management

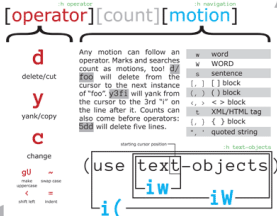
**:split** (or **:sp**) split window horizontally  
**:vsplit** (or **:vs**) split window vertically  
**:new** open a new window horizontally  
**:vnew** open a new window horizontally  
**Ctrl-w** switch cursor between windows  
**Ctrl-w r** rotate window positions

### Shell Command

**:shell** Open separate command prompt  
**exit** Exit command prompt

### Customization

**:set number** Show line numbers  
**:set tabstop=<N>** Change tab width  
**:set hlsearch** Highlight search matches  
**:set autoindent** Enable auto indentation



**gg** first line

**^b** up 1 page

**^u** up n page

**k** up n line

**j** down 1 line

**^d** down 1 page

**^f** down n page

**G** last line

**l** next character

**e** end of word

**w** beginning of next word

**E** end of WORD

**W** end of WORD

**\$** end of line

**:h cmd** Normal mode **cmd** help

**:h i\_cmd** Insert mode **cmd** help

**:h v\_cmd** Visual mode **cmd** help

**:h c\_cmd** Command-line editing **cmd** help

**:h :cmd** Command-line **cmd** help

**:h 'option'** **option** help

**:helpgrep** Search through all help docs!



**<CR>**  $\Lambda$ m  $\backslash$ r Enter

**<Tab>**  $\Lambda$ i  $\backslash$ t Tab

**<C-n>**  $\Lambda$ n  $\backslash$ Ctrl-n

**<M-n>**  $\Lambda$ b-n

**<Esc>**  $\Lambda$ [ Escape

**<BS>**  $\Lambda$ h  $\backslash$ b Backspace

**<Del>** Delete



**SEARCHING**

Prev	Next	Forward	Backward	Matches
N	n	/foo	?foo	foo
*	*	#	#	word under cursor
x	x	Tx	upto X	
;	;	fx	Fx	find X

**mm** set mark # (a-z) in file

**mM** set mark # (A-Z) across files

**'** jump to first char of last-changed text

**''** jump to first char of line containing #

**''** jump to exact character #

**''** jump back to last jump



**p** paste after cursor

**u** undo

**gf** first file under cursor in path and jump to it

**x** delete character after cursor

**ng** jump to line n

**zz** center screen on cursor

**==** auto-indent current line

**P** paste before cursor

**^r** redo

**dd** delete current line

**%** jump to matching paren

**zo** jump back

**zt** align top of screen with cursor

**<<** shift current line left by shiftwidth

**^[]** return to Normal mode

**.** repeat

**yy** yank current line

**r** replace char under cursor

**i** jump forward

**z** align bottom of screen with cursor

**zb** align bottom of screen with cursor

**>>** shift current line right by shiftwidth

7 words

<http://www.vimcheatsheet.com>

1 WORD

**:set opt?** View current value of **opt**

**:set noopt** Turn off flag **opt**

**:set opt** Turn on flag **opt**

**:set opt=val** Overwrite value of **opt**

**:set opt+=val** Append to value of **opt**

**:echo &opt** Access **opt** as a variable

**hidden** hid Lets you switch buffers without saving

**laststatus** ls Show status line never (0), always (1) or with +2 windows (1)

**hlsearch** hls Highlight search matches. Also see 'highlight'

**number** nu Show line numbers

**showcmd** sc Show commands as you type them

**ruler** ru Show line and column number of the cursor

**backspace** bs Set to '2' to make backspace work like some editors

**wrap** Control line wrapping

**background** bg Set to 'dark' if you have a dark color scheme

Use **§** instead of **f** when beginning text-object motions to include delimiters or surrounding whitespace. For example, **§f** will change "foo" into "f"; but **df** will delete the parentheses as well.

**ENTERING INSERT MODE**

beginning of line **I** before cursor **i** after cursor **a** end of line **A**

previous line **O** next line **o** substitute character **s** substitute line **S** line from cursor **C**

**ENTERING VISUAL (SELECT) MODE**

**v** switch cursor to start/end

**gv** re-select previous area

**I** prepended to each Visual block line

**o** jump to start of prior area

**<** jump to start of prior area

**COOL INSERT MODE STUFF**

**^w** delete word before cursor

**^r** insert the contents of register **r**

**^t** increase line indent by shiftwidth

**^x^l** line completion

**^u** delete line before cursor

**^R** use the expression register (try **^R**)

**^d** decrease line indent by shiftwidth

**^n** find next completion suggestion according to complete

**COMMAND-LINE MODE ONLY**

**^f** insert word under cursor

**^r^w** completion suggestions

**^d** completion suggestions

Put **set complete=^f,^r,^w** in your **vimrc** so you can type **^f** in Command-line mode to refer to the directory of the current file, regardless of **pwd**.

Supply **%** as a range to the **substitute** command to run it on every line in the file. **%s/Scr Ibb1/Des Igrn/** "Scribble" -> "Despined"

Specify the "g" flag to apply the substitution to every match on a line. **%s/[dia]/g** "baddy" -> "by"

Vim supports many regular expression features.

**:/** **/k/ax/** "book" -> "Max"

Use **///** instead of **/** if you want to search across multiple lines. **%s/heat/\_#Bungle/ant/>** "CheatsheetBungler" -> "Cantor"

Special escapes can be used to change the case of substitutions. **:%s/(f\.\.)\_W/V/E\_** "foobar" -> "FOOBAR"

Use **[global]** to perform a command on matching lines. **:/g/foobar/d** Delete all lines containing "foobar"

If your pattern contains slashes, just use a different character as your delimiter. **:%s/Data/Lore, Brent, Spliner.**

Use **///** to evaluate expressions with replacement groups. **:%s,\d\_\\$submatch(0) + 1,g** "10 25" -> "21 36"

**:ls** List all open files

**:b path** Jump to unique file matching **path**. Use **<Tab>** to scroll through available completions!

**:bn** Jump to file **n**, number from first column of **:ls**

**:bnext** Jump to next file

**:bprev** Jump to previous file

**:bdelete** Remove file from the buffer list

**:edit** Open a file for editing

**:enew** Open a blank new file for editing

**REGISTERS & CLIPBOARDS**

All commands that delete, copy, or paste text use registers. To change which register is used by a command, type the register before the command. The default register is called "the unnamed register", and it is invoked with a pair of double-quotes (**""**). Typing **""** or **''** is the same as typing **""** or **''**. Think of the first **''** as a short way of saying "register", so **''** is pronounced "register", and **''** "register".

**:registers** View all current registers

**:echo @r** Access register **r** as a variable

**"/** Last search pattern register Contains the last pattern you searched for

**"\_** The black hole register Use this to delete without clobbering any register (**dd**)

**"0** Last yank register Contains the last text you yanked

**"1** Last big delete register Contains the last line(s) you deleted

**"2-9** Big delete register stack Every time **"1** is written to, its content is pushed to **"2**, then **"2** to **"3**, and so on

**"\_** Small delete register Contains the last text you deleted within a single line

**+** System clipboard If the OS integration gods smile upon you, this register reads and writes to your system clipboard.

**"a-z** Named registers 26 registers for you to play with

**"A-Z** Append registers Using upper-case to refer to a register will append to it rather than overwrite it

**qr** Record Record into register **r**. Stop recording by hitting **q** again

**@r** Playback Execute the contents of register **r**

**@@** Repeat last playback Repeat the last **@r**; this is particularly useful with a count.

Use **§** instead of **f** when beginning text-object motions to include delimiters or surrounding whitespace. For example, **§f** will change "foo" into "f"; but **df** will delete the parentheses as well.

**ZZ** Write current file, if modified, and quit

**ZQ** Quit without checking for changes (like **q!**)

**:write** Write current file

**:wq** Write current file and quit

Use **scriptnames** to list all files sourced during initialization.

**:syntax** Enable and configure syntax highlighting. Use **:syntax toprule** to redraw broken highlights

**:make** Run a compiler and enter quickfix mode

**!** Execute external shell command

**!** Filter motion with shell command

Use **earlier** and **later** to quickly jump backward and forward in a file's history.

**:read** Read external program output into current file

## Modes & Controls

**Command Mode** ESC (commands preceded by :)

**Insertion Mode** Entered on insertion or change

### Starting VI (command line)

<b>vi &lt;filename&gt;</b>	Edit <i>filename</i>
<b>vi -r &lt;filename&gt;</b>	Edit last version of <i>filename</i> after crash
<b>vi +n &lt;filename&gt;</b>	Edit <i>filename</i> at line <i>n</i>
<b>vi + &lt;filename&gt;</b>	Edit <i>filename</i> at end of file
<b>vi +/str &lt;filename&gt;</b>	Edit <i>filename</i> at first occurrence of <i>str</i>

In insertion mode the following should be preceded by ESC:

<b>:w</b>	Save
<b>:x</b>	Save & Exit
<b>:q</b>	Exit if no changes made
<b>:q!</b>	Exit & discard any changes

## Cursor Navigation

<b>h</b> or ◀	Cursor left
<b>j</b> or ▼	Cursor down
<b>k</b> or ▲	Cursor up
<b>l</b> or ▶	Cursor right
<b>w</b>	Next word
<b>W</b>	Next blank delimited word
<b>b</b>	Start of word
<b>B</b>	Start of blank delimited word
<b>e</b>	End of word
<b>E</b>	End of blank delimited word
<b>(</b>	Back a sentence
<b>)</b>	Forward a sentence
<b>{</b>	Back a paragraph
<b>}</b>	Forward a paragraph
<b>0</b>	Beginning of line
<b>\$</b>	End of the line
<b>1G</b>	Start of file
<b>G</b>	End of file
<b>:n</b>	<i>n</i> th line of file
<b>f&lt;char&gt;</b>	Forward to <i>char</i>
<b>F&lt;char&gt;</b>	Back to <i>char</i>
<b>H</b>	Top of screen
<b>M</b>	Middle of screen
<b>L</b>	Bottom of screen
<b>%</b>	Matching bracket
<b>gg</b>	Start of document

## Inserting Text

<b>i</b>	Insert before cursor
<b>a</b>	Append after cursor
<b>I</b>	Insert before line
<b>A</b>	Append after line
<b>o</b>	Add new line after current line
<b>O</b>	Add new line before current line
<b>r</b>	Overwrite one character
<b>R</b>	Overwrite many characters
<b>:r &lt;file&gt;</b>	Reads <i>file</i> and inserts it after this line
<b>p</b>	Put after the position or line
<b>P</b>	Put before the position or line
<b>C</b>	Rewrite the whole line

## Deleting Text

<b>x</b>	Delete character to right of cursor
<b>X</b>	Delete character to left of cursor
<b>D</b>	Delete the rest of line
<b>dd</b> or <b>:d</b>	Delete current line
<b>ndw</b>	Deletes the next <i>n</i> words
<b>ndb</b>	Deletes the previous <i>n</i> words
<b>ndd</b>	Deletes <i>n</i> lines starting with current
<b>:x,yd</b>	Delete lines <i>x</i> through <i>y</i>
<b>:r &lt;file&gt;</b>	Reads <i>file</i> and inserts it after this line
<b>d{nav_cmd}</b>	Overwrite many characters
<b>:r &lt;file&gt;</b>	Reads <i>file</i> and inserts it after this line

## Searching

<b>/string</b>	Search forward for string
<b>?string</b>	Search backwards for string
<b>n</b>	Go to next match
<b>N</b>	Go to previous match
<b>:set ic</b>	Ignore case while searching
<b>:set noic</b>	Case-sensitive searching
<b>:set nu</b>	Turn on line numbers
<b>:x,yg/str</b>	Search for <i>str</i> from line <i>x</i> to line <i>y</i>
<b>:g/str/cmd</b>	Run <i>cmd</i> on lines containing <i>str</i>
<b>*</b>	Search for next instance of current word
<b>#</b>	Search for last instance of current word

## Replacing

<b>:s/pt/str/flag</b>	Replace pattern with string
<b>Flags</b>	
<b>g</b>	Replace all occurrences of pattern
<b>c</b>	Confirm replaces
<b>&amp;</b>	Repeat last <b>:s</b> command

## Other

<b>u</b>	Undo last change
<b>J</b>	Join lines
<b>nJ</b>	Join next <i>n</i> lines
<b>.</b>	Repeat last command
<b>U</b>	Undo all changes to line
<b>:N</b>	Open split screen
<b>v</b>	Visual mode
<b>ctrl + c</b>	Escape insert mode

