



第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

# 第一季: Vim 原生操作总结

Jinghui Hu

<2024-01-23 Tue>



第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

B 站个人空间: <https://space.bilibili.com/1969478249>





第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

# Shortcuts 快捷键



# Map 的作用

第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

## 1 修改按键原本的意思

```
map <C-S> :w<CR>
imap <F3> <Esc><Right>n<Insert>
```

## 2 执行特定的函数

```
map <F2> a<C-R>=strftime("%c")<CR><Esc>
```

- <C-R>= 进入 Ex 命令
- strftime("%c") 执行函数，获取当前时间

## 3 查看键位绑定 :map

```
:map
x  gx          <Plug>NetrwBrowseXVis
n  gx          <Plug>NetrwBrowseX
x  <Plug>NetrwBrowseXVis * :<C-U>call netrw#BrowseXVis()<CR>
```



# Map 使用的注意点

## 1 两种不同风格的键位绑定

- map
- noremap 避免递归

## 2 取消键位绑定

- unmap

## 3 各种模式下的 Mapping

模式	map
normal	nmap, map
visual	vmap, map
insert	imap, map!
command-line	cmap, map!
operator-pending	omap, map

第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声



# Abbreviation 简写

第一季: Vim 原生

操作总结

Jinghui Hu

Shortcuts 快捷键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

## 1 abbr 可以用在不同模式下

- insert, replace 和 Ex command line

## 2 修复拼写错误

```
iab tihs this
```

## 3 简写加快 insert 模式下的输入

```
iab eml yourname@example.com
```

```
iab #i #include
```



第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

# Window, Buffer 和 Tab



# Window, Buffer 和 Tab

- 1 Buffer 是编辑文件的内存副本 (copy)
- 2 Window 是 Buffer 的一个视口 (viewport)
- 3 Tab 承载着 Window, Tab 是可以包含多个 Window

The screenshot shows the Vim editor interface. The menu bar at the top includes File, Edit, Tools, Syntax, Buffers, Window, and Help. Below the menu bar is a toolbar with various icons. The main editing area displays the following content:

```
1. Introduction windows-intro window

Summary:
  A buffer is the in-memory text of a file.
  A window is a viewport on a buffer.
  A tab page is a collection of windows.

A window is a viewport onto a buffer. You can use multiple windows on one
buffer, or several windows on different buffers.

A buffer is a file loaded into memory for editing. The original file remains
unchanged until you write the buffer to the file.
windows.txt [Help][R0] 38,41 2%
```

At the bottom of the screen, the status bar shows:

```
[No Name] 0,0-1 All
:h windows
```

第一季: Vim 原生

操作总结

Jinghui Hu

Shortcuts 快捷键

Window, Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声





# Buffer

1 Buffer 三种状态: hidden, active, inactive

2 查看 buffer 列表 :ls 或者 :buffers 或者 :files

```
:ls
```

```
1 %a      "e01.org"                line 1
2          "e02.org"                line 0
```

■ Indicators: % Buffer 在当前 Window, a=active, h=hidden

■ 更多细节查看 :h :ls

3 跳转 buffer 列表, 上面列表的第一列的数字为 <buffer\_id>

```
:b <buffer_id>
```

```
:bn/:bp 前一个/后一个
```

4 清理 Buffer

■ :bdelete 卸载 Buffer

■ :bwipeout 卸载并且删除 Buffer



# Window

第一季: Vim 原生

操作总结

Jinghui Hu

Shortcuts 快捷键

Window, Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

- 1 水平分屏 `:split`, CTRL-W s
- 2 垂直分屏 `:vsplit`, CTRL-W v
- 3 新建空的 Buffer `:new`, CTRL-W n, `:vnew`
- 4 其它命令 `:w` 和 `:q`
- 5 为每个文件打开一个 Window  
`vim -o file1 file2 file3`
- 6 Window 跳转命令 CTRL-W w, CTRL-W W, CTRL-W CTRL-W
- 7 更多参考 `:help Q_wi`, `:help windows`



# Tab

第一季: Vim 原生

操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

- 1 Tab 是 GUI 编辑器的一种常见的文件管理形式, 在 vim7 以后引入
- 2 使用 Tab 打开文件 `:tabe file`
- 3 查看所有 Tab `:tabs`
- 4 关闭 Tab `:tabc`
- 5 循环 Tab 列表 `:tabn`, `:tabp`
- 6 在新 Tab 中打开光标下的文件  
CTRL-W gf
- 7 `:help tabpage`



第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

# Visual 模式



# 不同的 Visual 模式

第一季: Vim 原生

操作总结

Jinghui Hu

Shortcuts 快捷键

Window, Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

Visual 模式提供了对文本可视化批量操作

- 1 不同的维度的 Visual 模式
  - visual block: `c-v` , 俗称: “块选”
  - visual line: `v`
  - visual char: `ˆv`
- 2 鼠标操作支持, 设置 `mouse` 选项 `'n'` 或者 `'a'`
- 3 `d` 删除选中, `y` 复制选中
- 4 `I` 按块插入, `A` 按块追加
- 5 `:h visual`



■ 只不过 Select 选中过后按键就直接修改文本

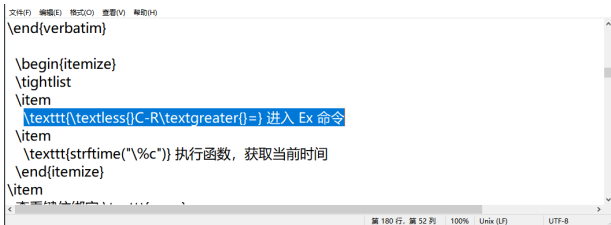
- Select 更像 notepad 的选中操作

## 2 使用 C-g 来切换 Visual 和 Select

3 gv 之前选中的文本

#### 4 < 选中文本的左端点

### 5 > 选中文本的右端点





第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

## 更多跳转技巧



# jumplist: 跳转列表

第一季: Vim 原生

操作总结

Jinghui Hu

Shortcuts 快捷键

Window, Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

## 1 通过 :ju 查看跳转列表

```
jump line  col file/text
      3      1      0 some text
      2     70      0 another line
      1   1154     23 end.
>
```

## 2 更多操作: :h :jumps

CTRL-I 往后跳转  
CTRL-O 往前跳转





# tagstack: TAG 栈

第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

- 1 准备 TAGS 文件, 设置 tags 选项, :h 'tags'

```
ctags -R src
```

- 2 通过 :tags 查看 tag 栈

```
# TO tag          FROM line  in file/text
1  1 main          1  harddisk2:text/vim/test
> 2  2 FuncA        58  i = FuncA(10);
3  1 FuncC         357  harddisk2:text/vim/src/amiga.c
```

- 3 更多操作 :h tagstack

CTRL-] 进入 TAG

CTRL-T 往上弹出栈



# quickfix: 快速修改列表

第一季: Vim 原生

操作总结

Jinghui Hu

Shortcuts 快捷键

Window, Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

- 1 执行 `:make` 编译后产生的列表, 用于快速修改错误
- 2 查看 `quickfix` 列表 `:cl`
- 3 跳转 `quickfix` 列表 `:cc` 和 `:ll`
- 4 列表的跳转指令 `:cn` 和 `:cp`

```
:cl
```

```
4 source/anlys.c:20 col 29: error: 'block_node_t' {aka 'struct...
9 source/anlys.c:95 col 37: error: 'block_node_t' {aka 'struct...
14 source/anlys.c:137 col 37: error: 'block_node_t' {aka 'struc...
Press ENTER or type command to continue
```



# 其它列表： changelist, 使用 mark

第一季: Vim 原生

操作总结

Jinghui Hu

Shortcuts 快捷键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

## 1 changelist

- g; 跳转到前一个修改
- g, 跳转到后一个修改
- :changes 查看修改列表

## 2 marks

- m<reg> 设置 mark
- '<reg> 跳转到制定 mark
- <reg> 小写仅单文件, 大写跨文件



# 鼠标支持

第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

- 1 `set mouse=n/v/i/c/h/a/r/A`
  - `set mouse=nvi` 开启 Normal, Visual 和 Insert 模式的鼠标支持
  - `set mouse=a` 开启所有模式的鼠标支持
- 2 `:h mouse`





第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

**重复操作**

未分类

额外的常用操作

尾声

# 重复操作



# Repeat - 快速重复之前的编辑命令

第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

- 1 . 重复上一个修改
- 2 @: 重复上一个命令
  - :ls 查看当前的所有 Buffer
  - 一顿编辑之后
  - @: 继续执行之前的 Ex 命令, 重复列举所以 Buffer
- 3 :help single-repeat



# Record - 录制宏

宏 (macro) 指的是记录按键的次序, 然后重放记录后的按键次序

## 1 录制宏

- q<reg> 开始录制宏, 并记录到寄存器 <reg> 中

- q 结束录制宏

## 2 @<reg> 唤醒寄存器 <reg> 中的宏

## 3 @@ 唤醒最近一次执行的宏





# Global - 选取然后执行命令

1 :<range>g/<re>/<cmd>

- <range> 文件的影响范围: 1,5 第 1 到 5 行
- g global 命令的简写, 寻找匹配 <re> 正则表达式的行
- <cmd> 执行命令

2 反向选中, 可以使用 g! 或者 v

举个例子: 排序 css 花括号里面的属性 KV 对

```
body {  
    color: black;  
    background: white;  
}
```

```
:g/{/ .+1,/}/-1 sort
```

- g/{/ 查找 { 行
- .+1,/}/-1 在当前行 . 的下一行和继续查找 } 花括号范围内
- 在上述范围内进行排序 sort

第一季: Vim 原生

操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声





第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

未分类



# 寄存器

第一季: Vim 原生

操作总结

Jinghui Hu

Shortcuts 快捷键

Window, Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

寄存器用来存编辑中的数据：删除，复制和宏

- 1 a-z named register, 用于记录显示制定的操作
- 2 A-Z 追加到对应的小写寄存器中
- 3 \_ 黑洞寄存器, 用于忽略无用的文本
- 4 :reg 查看所有寄存器, :reg a 查看寄存器 a
- 5 <reg> <action> 带寄存器的操作
  - "ryiw yiW 到寄存器 r 中
  - "Ryiw yiW 并追加到寄存器 r 中
- 6 :h reg



# 折叠

第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

- 1 折叠设置: 'foldmethod'
  - manual 手工创建
  - indent 基于代码缩进, 例如: Python
  - marker 基于 foldmarker
- 2 折叠快捷键 :h Q\_fo
  - zf 创建折叠 (Fold)
  - zo/zO 打开折叠 (Open)
  - zc/zC 关闭折叠 (Close)

## 一些使用案例

```
set foldmethod=manual/indent/marker
set foldmarker=,
set foldmarker=<,>
```



# 过滤器 Filter

第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

- 1 `<range>! <filter-cmd>` 选取范围过滤
- 2 `<selection>! <filter-cmd>` 针对选中的区域过滤
- 3 `:h complex-change`





第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

## 额外的常用操作



# 常用设置

第一季: Vim 原生

操作总结

Jinghui Hu

Shortcuts 快捷键

Window, Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

- 1 `:set hls` 高亮显示搜索的字符串 `'hlsearch'`
- 2 `:set is` 立即反馈搜索 (in-place search) `'incsearch'`
- 3 `:set nu` 设置行号 `'number'`
- 4 `:set hidden` 设置隐藏
  - Buffer 未保存时, 使用 `e` 编辑其它文件, 不需要 `e!`
- 5 `:set scrolloff=5` 光标上下始终有几行文本

`:set scrolloff?`



# 杂项常用命令

第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

- 1 `ga` 现在当前光标下字母的 ASCII 数值
- 2 `gf` 打开当前光标下的文件
- 3 `<Ctrl-g>` 显示当前文件的文件名全路径
- 4 `g<Ctrl-g>` 显示当前文件的字数
- 5 `<<` 向左缩进代码
- 6 `>>` 向右缩进代码
- 7 `:!cmd` 执行 shell 命令
- 8 `vimdiff` 使用 vim 的 diff 工具



第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

# 尾声





# 第一季总结

第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

第一季所有视频及相关材料 Github 仓库 <sup>1</sup>

- 1 Vim 介绍、七个习惯
- 2 Vim 发展及其帮助系统
- 3 vimtutor 重温 Vim 的基础操作
- 4 VimGolf 打磨你的操作技巧
- 5 Vim 9.1 发布及 vim 编译安装
- 6 Text Object
- 7 救你于水深火热的一些 vim 概念
- 8 薛定谔的 cat
- 9 QuickRef
- 10 键盘发展及编辑器学习曲线

---

<sup>1</sup><https://github.com/Jeanhwea/Vim-Tips>



# 高端的食材往往只需要简单的烹饪

第一季: Vim 原生  
操作总结

Jinghui Hu

Shortcuts 快捷  
键

Window,  
Buffer 和 Tab

Visual 模式

更多跳转技巧

重复操作

未分类

额外的常用操作

尾声

