

# Vim

----创新部于海威

# 目录

- ◎ 为什么分享
- ◎ Vim实例秀
- ◎ Vi基础（三种模式下的基础命令）
- ◎ Vi进阶（进阶命令、插件使用、快捷键配置）
- ◎ Vi自身优劣势

# 为什么分享

- ◎ 每个开发人员都有使用vi的场景：比如在我们测试机上进行一些配置时
- ◎ 某些场景结合使用vi会更方便
- ◎ 对它多了解一些，也许就会喜欢上用它
- ◎ 有同事已经表现出兴趣

# 几个真实场景

- ◎ 最近后端代码重构，static变量变为const
- ◎ 婷婷批量生成sql
- ◎ 台湾行政区快速声明为php数组

# 基础命令

- ◎ 三种模式

普通、编辑、末行

# 三种模式切换

- ◎ 普通→编辑 <i>,<l>,<a>,<A>,<o>,<O>
- ◎ 普通→末行 <:>
- ◎ 编辑→普通 <esc>
- ◎ 末行→普通 <esc>

# 普通模式

- 移动:

<hijkl>左下上右

<0>行首

<\$>行尾

<gg>首行

<G>尾行

<w>或<e>下一词语、

- 滚屏

<ctrl+b> 上滚一屏

<ctrl+f> 下滚一屏

<ctrl+u> 上滚半屏

<ctrl+d> 下滚半屏

# 普通模式

- 删除:

<x>单字符

<daw>当前词语

<de><dw>从当前字符开始删除当前词的后  
面部分

<dd>行

- 替换: <r>然后输入新字符, 标示把当前  
字符替换为新字符



# 普通模式

- ◎ 复制粘贴

<yy> 复制当前行

<p> 粘贴到后面

<P> 粘贴到前面

- ◎ redo undo

<u> 撤销

<ctrl+r> 重做

# 编辑模式

- ◎ <i> 字符前编辑
- ◎ <l> 行首编辑
- ◎ <a> 字符后编辑
- ◎ <A> 行尾编辑
- ◎ <o> 在后面新起一行编辑
- ◎ <O> 在前面新起一行编辑

# 末行模式

- ◎ :w 保存编辑
- ◎ :q 关闭当前窗口
- ◎ :q! 强制关闭
- ◎ :wq 先保存再退出
- ◎ :set nu 显示行号
- ◎ :set nonu 不显示行号
- ◎ :set paste 粘贴（格式友好）
- ◎ :set readonly 设置只读模式

# 进阶

- ◎ 命令（大小写转换、大括号匹配、删除块、正则替换等）
- ◎ 插件配置（Project、cscope、bufferexplorer）
- ◎ 快捷键配置，DIY

# 进阶命令

- ◎ 代码风格自动调整：选中后 <=>

- ◎ 括号对()\[\]\{}操作：

  - <%> 来回跳转

  - 移到左或右括号上<d%>：删除括号及其中内容

  - 移到左或右括号上<y%>：复制括号及其中内容

# 进阶命令

## ◎ 选择：

<v> 选中当前字符

<shift+v> 选中当前行

<ctrl+v> 进行矩形文本块选择

上面三种进行选择方式后，通过上下左右键来进一步作选择

选择后，可以用过<y><d>等命令对选择的内容进行复制删除操作

# 进阶命令

## ◎ 大小写转换：

<guaw> 当前词语转为小写

<gUaw> 当前词语转为大写

对选中内容<gu> 选中区变为小写

对选中内容<gU> 选中区变为大写

## ◎ 引号对

在单引号里<di'> |<yi'>分别表示删除|复制  
当前单引号里面的内容

双引号类似

# 进阶命令

## ◎ 替换

:s/old/new 对当前行进行一次替换

:s/old/new/g 对当前行或选中范围进行全部替换

:%s/old/new 对每一行的第一个进行替换

:%s/old/new/g 对全文进行全部替换



# 进阶命令

## ◎ 分屏

:sp 上下分屏打开当前文件

:vsp 左右分屏打开当前文件

<ctrl+w>然后<s> 同 :sp

<ctrl+w>然后<v> 同:vsp

## ◎ 当前屏尺寸调整

<ctrl+w> 然后 <o> 最大化当前屏

<ctrl+w> 然后 <+> 增加当前屏高度

<ctrl+w> 然后 <-> 减少当前屏高度

<ctrl+w> 然后 <=> 所有屏均分高度

# 进阶命令

- ◎ 屏间跳转

<ctrl+w> 然后 <h或j或k或l> : 跳转到左  
|下|上|右屏

两次<ctrl+w> 顺时针跳转屏幕

- ◎ 代码自动提示补全

编辑模式<ctrl+n>

# 插件配置

- ◎ 下载插件的文件到~/.vim/目录下doc和plugin子目录中

详见：<http://yuhaiwei.888.blog.163.com/blog/static/619990182014727238733/>

# 自定义命令

- ◎ ~/.vimrc文件来定义自己的快捷键

# 优势

- ◎ 轻量级
- ◎ 多项目方便（每个项目只是一个终端页签）
- ◎ 所有linux自带
- ◎ 正则强大，这一点就是一个好的工具
- ◎ 速率

脱离鼠标

省去IDE和终端的切换（git, svn, php  
语法检查等）

# 劣势

- ◎ 背诵各种命令，烦人
- ◎ 配置过程稍显复杂，人工参与多