

Gvim编辑器使用



代码编写速度提高10倍以上!!!

- 1.gvim快捷键的操作
- 2.函数补全
- 3.snipMate代码补全

姓名:李强强

QQ:447096447

邮箱:liqiangqiang@lampbrother.net



本节目标

- 1.gvim软件介绍
- 2.gvim配置文件
- 3.gvim自动缩进
- 4.gvim页面编码
- 5.gvim函数补全功能
- 6.gvim snipMate极速补全



Vim/Vi 简介

vi是linux和unix下使用最为广泛的一种文本编辑器，最大的优势是它的快捷键操作，使的不用鼠标也可以快速操作文本

vim是vim的增强版本，比如语法和函数高亮显示等

gvim是window下的vim编辑器，而且使用更方便、快速

下载网址：<http://www.vim.org/download.php>



version 1.1
April 1st, 06
翻译: 2006-5-22

vi / vim 键盘图

Esc
命令
模式

~ 转换 大小写	! 外部 过滤器	@ 运行 宏	# prev ident	\$ 行末	% 括号 匹配	^ "软" 行首	& 重复 :s	* next ident	(句首) 下一 句首	"soft" bol down	+ 后一行 行首
\. 跳转到 标注	1	2	3	4	5	6	7	8	9	0 "硬" 行首	- 前一 行首	= 自动 ³ 格式化
Q 切换到 ex 模式	W 下一 单词	E 词尾	R 替换 模式	T back 'till	Y 拷贝 行	U 撤销 行命令	I 到行首 插入	O 分段 (前)	P 粘贴 (前)	{ 段首	}	段尾
q 录制 宏	w 下一 单词	e 词尾	r 替换 字符	t 'till	y 拷贝 ^{1,3} 行	u 撤销 命令	i 插入 模式	o 分段 (后)	p 粘贴 (后)	[杂项]	杂项
A 在行末 附加	S 删除行 并插入	D 删除 至行末	F 行内字符 向前查找	G 文尾/ 行号	H 屏幕 顶行	J 合并 两行	K 帮助	L 屏幕 底行	: ex 命令	" 寄存器 ⁴ 标识	行首/ 列	
a 附加	s 删除字符 并插入	d 删除 ^{1,3}	f 行内字符 查找	g 附加 ⁶ 命令	h ←	j ↓	k ↑	l →	: 重复 ; u/T/f/F	' 跳转到标 注的行首	\ 未使用!	
Z 退出 ⁴	X 退格	C 修改 至行末	V 可视 行模式	B 前一 单词	N 查找 上一处	M 屏幕 中间行	< 反退格 ³	> 退格 ³	? 向前 搜索			
Z 附加 ⁵ 命令	x 删除 (字符)	c 修改 ^{1,3}	v 可视 模式	b 前一 单词	n 查找 下一处	m 设置 标注	, 反向 ; u/T/f/F	. 重复 命令	/ 向后 搜索			

动作 移动光标, 或者定义操作的范围

命令 直接执行的命令,
红色命令进入编辑模式

操作 后面跟随后表示操作范围的指令

extra 特殊功能,
需要额外的输入

q 后跟字符参数

w, e, b 命令

小写(b): quux(foo, bar, baz);

大写(B): QUUX(foo, bar, baz);

主要ex命令:

:w (保存), :q (退出), :q! (不保存退出)

:e f (打开文件 f),

:%s/x/y/g ('y' 全局替换 'x'),

:h (帮助 in vim), :new (新建文件 in vim)

其它重要命令:

CTRL-R: 重复 (vim),

CTRL-F/-B: 上翻/下翻,

CTRL-E/-Y: 上滚/下滚,

CTRL-V: 块可视模式 (vim only)

可视模式:

漫游后对选中的区域执行操作 (vim only)

备注:

(1) 在 拷贝/粘贴/删除 命令前使用 "x (x=a..z,")
使用命令的寄存器("剪贴板")

(如: "ay\$ 拷贝剩余的行内容至寄存器 'a')

(2) 命令前添加数字
多遍重复操作
(e.g.: 2p, d2w, 5i, d4j)

(3) 重复本字符在光标所在行执行操作
(dd = 删除本行, >> = 行首缩进)

(4) ZZ 保存退出, ZQ 不保存退出

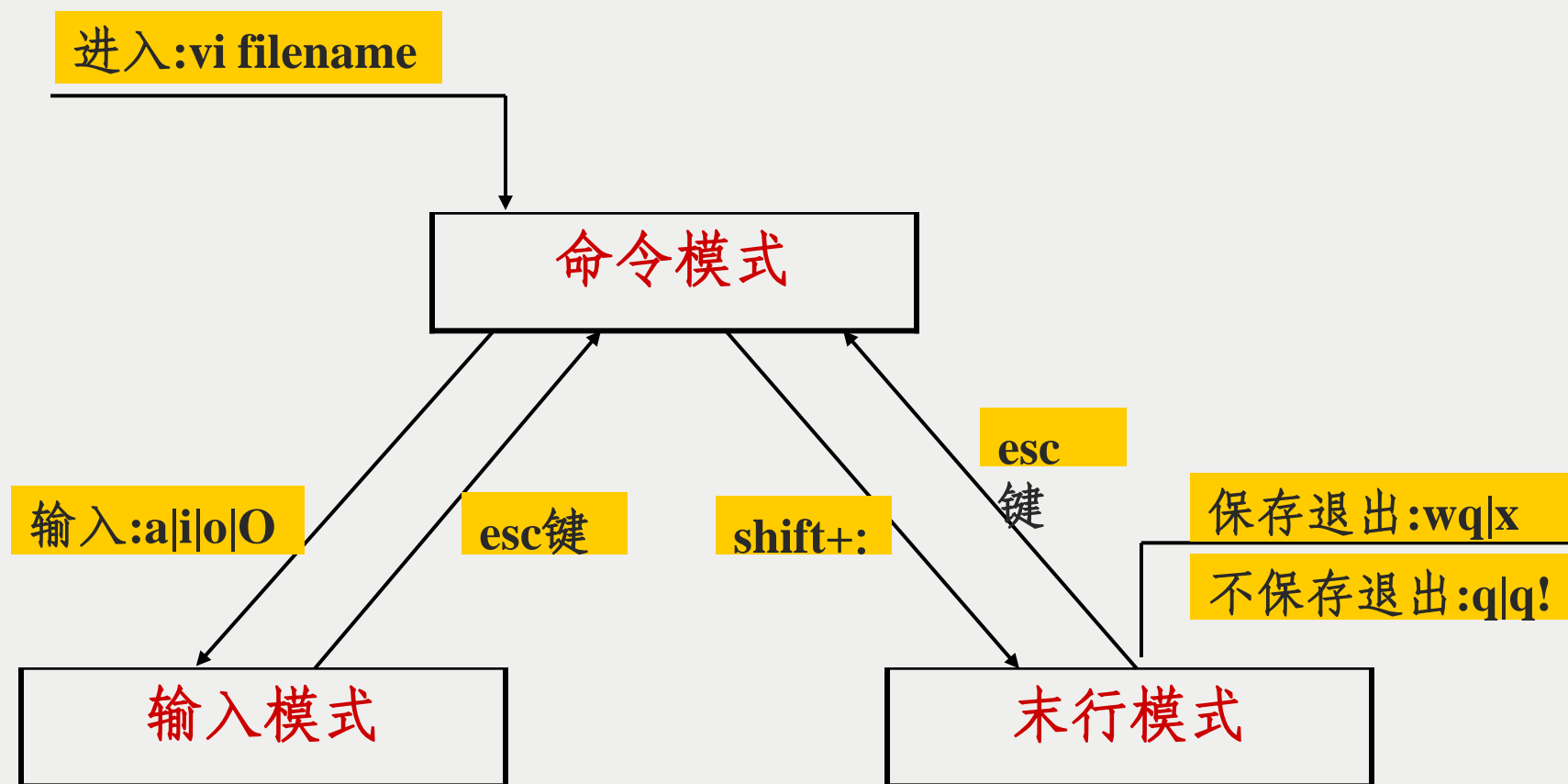
(5) zt: 移动光标所在行至屏幕顶端,
zb: 底端, zz: 中间

(6) gg: 文首 (vim only),
gf: 打开光标处的文件名 (vim only)

原图: www.viemu.com 翻译: fdl (linuxsir)



Gvim工作模式





命令模式

命令	作用
h	向左移动
j	向下移动
k	向上移动
l	向右移动
yy 2yy	复制行
dd 2dd	删除、剪切行
p	粘贴到下一行
P	粘贴到上一行
^ \$	定位到行首 行尾
dG	从当前行删到文档最后一行



命令模式=>插入模式

命令	作用
a	在光标之后插入
i	在光标之前插入
o	在下行插入
O	在上行插入



末行模式

命令	作用
<code>:x :wq</code>	保存退出
<code>:q</code>	不保存退出
<code>:q!</code>	强制不保存退出
<code>:!</code>	执行shell(dos)命令
<code>:r !</code>	把shell(dos)命令执行结果读入文档
<code>:%s/good/bad/g</code>	把good全部替换为bad
<code>:/good</code>	搜索good
<code>:G</code>	到最后一行
<code>:2</code>	定位到第2行



末行模式技巧操作

命令	作用
<code>:set nu</code>	设置行号
<code>:set nonu</code>	取消行号
<code>:1,10s/^/#/g</code>	在1到10行前加#注释
<code>:1,10s/#/^/g</code>	把1到10行前#注释取消
<code>:1,10>></code>	把1到10行向右移动两个tab
<code>:1,10<<</code>	把1到10行向左移动两个tab
<code>:1,10d</code>	把1到10行删除



命令模式=>替换模式

命令	作用
r	替换单个字符
R	替换多个字符
u	撤销上次操作
ctrl+y	恢复上次操作



gvim配置文件-1

"关闭兼容模式

```
set nocompatible
```

"模仿快捷键，如:Ctrl+A全选、Ctrl+C复制、Ctrl+V粘贴等等

```
source $VIMRUNTIME/vimrc_example.vim
```

```
source $VIMRUNTIME/mswin.vim
```

```
behave mswin
```

"gvim字体设置

```
set guifont=新宋体:h18:cGB2312
```

"gvim内部编码

```
set encoding=utf-8
```

"当前编辑的文件编码

```
set fileencoding=utf-8
```



gvim配置文件-2

"gvim打开支持编码的文件

```
set fileencodings=ucs-bom,utf-8,gbk,cp936,gb2312,big5,euc-jp,euc-kr,latin1
```

"set langmenu=zh_CN

"let \$LANG = 'zh_CN.UTF-8'

"解决console输出乱码

```
language messages zh_CN.utf-8
```

"解决菜单乱码

```
source $VIMRUNTIME/delmenu.vim
```

```
source $VIMRUNTIME/menu.vim
```

"设置终端编码为gvim内部编码encoding

```
let &termencoding=&encoding
```



gvim配置文件-3

"防止特殊符号无法正常显示

```
set ambiwidth=double
```

"缩进尺寸为4个空格

```
set sw=4
```

"**tab**宽度为4个字符

```
set ts=4
```

"编辑时将所有**tab**替换为空格

```
set et
```

"按一次**backspace**就删除4个空格了

```
set smarttab
```



gvim配置文件-4

"不生成备份文件,如index.html~

set nobackup

"开启行号标记

set number

"配色方案为desert

colo desert

"关闭上侧工具栏

set guioptions-=T

"关闭右侧滚动条

"set guioptions-=r

"开启自动缩进

set autoindent



js自动补全功能

The screenshot shows the GVIM1 editor window with a file named index.html. The code is as follows:

```
1 <html>
2   <body>
3     <font color="#ff0000">兄弟连</font>
4     <font color="#ff0000">兄弟连</font>
5     <font color="#ff0000">兄弟连</font>
6   </body>
7   <script type="text/javascript" charset="utf-8">
8     document.anchors
9   </script>
10 </html>
```

An auto-completion list is displayed for the text `document.anchors`. The list includes the following items:

- anchors
- applets
- childNodes
- embeds
- forms
- images
- links
- stylesheets
- body
- cookie
- documentElement
- domain
- lastModified
- referrer
- title
- URL

A yellow callout box on the right side of the editor contains the text: 快捷键: ctrl+x+o

At the bottom of the editor window, a status bar displays: 全能补全 (0~N~P) 匹配 1 / 57



php自动补全功能

index.php + (~\桌面) - GVIM1

文件(F) 编辑(E) 工具(T) 语法(S) 缓冲区(B) 窗口(W) 帮助(H)

```
1 mysql_affected_rows([resource link_identifier] | int
2
```

[Scratch] [预览] 1, 1 全部

```
1 <?php
2 mysql_affected_rows(
~ mysql_affected_rows( f
~ mysql_change_user( f
~ mysql_client_encoding( f
~ mysql_close( f
~ mysql_connect( f
~ mysql_create_db( f
~ mysql_data_seek( f
~ mysql_db_name( f
~ mysql_db_query( f
~ mysql_drop_db( f
~ mysql_errno( f
~ mysql_error( f
~ mysql_escape_string( f
~ mysql_fetch_array( f
~ mysql_fetch_assoc( f
~ mysql_fetch_field( f
~ mysql_fetch_lengths( f
index.p mysql_fetch_object( f
```

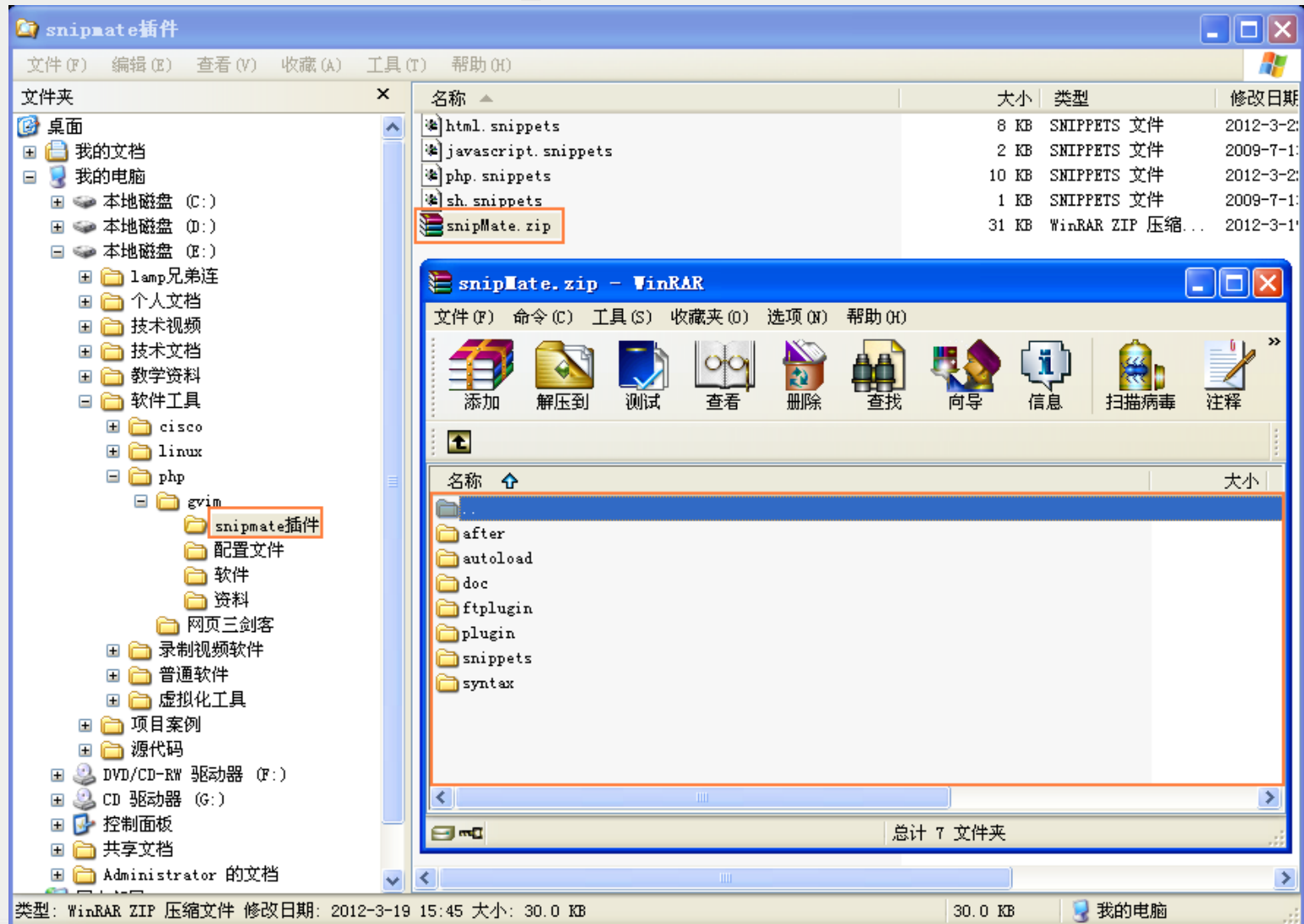
全能补全 (^O^N^P) 匹配 1 / 71

快捷键: ctrl+x+o

2, 5 全部

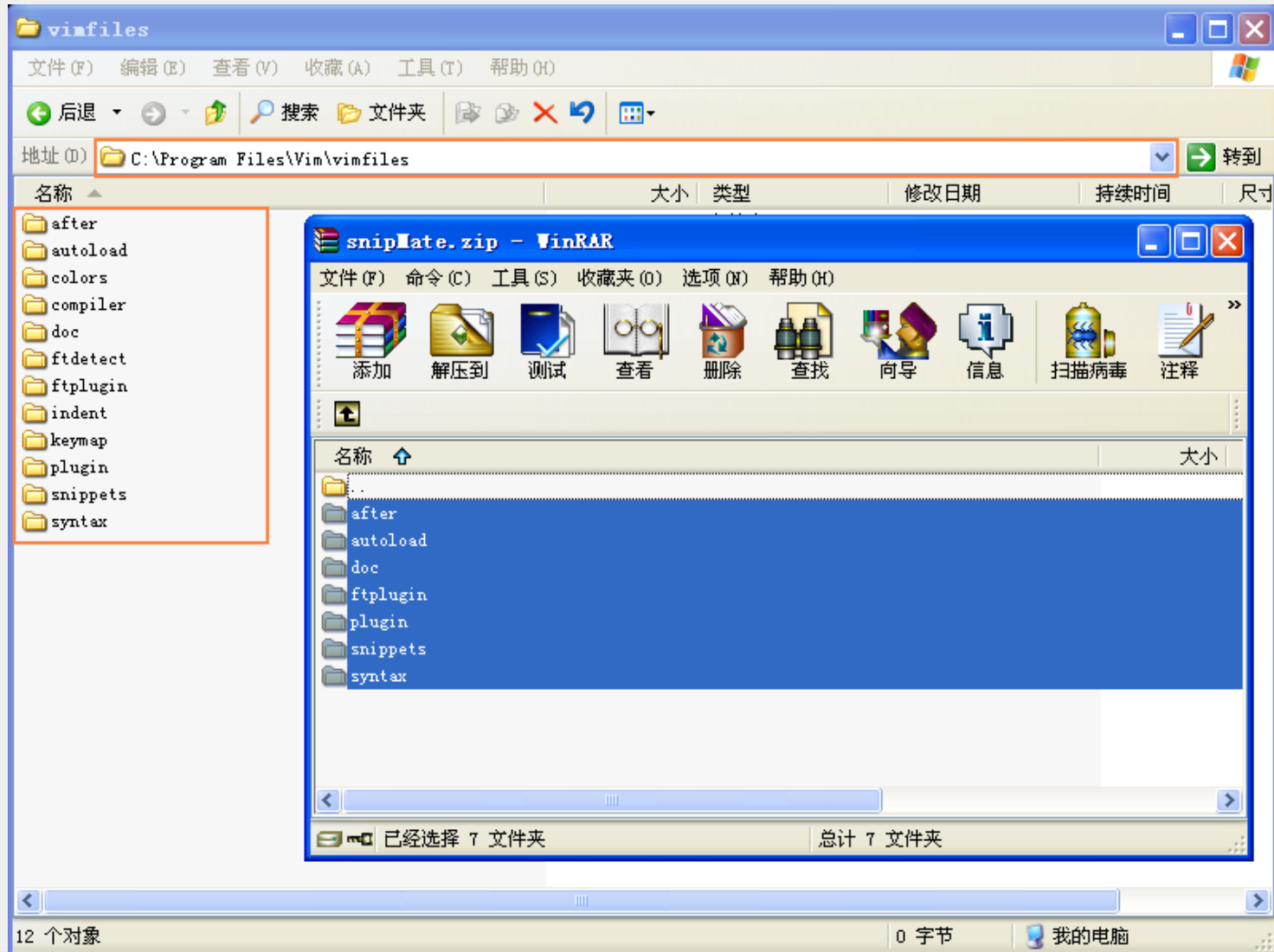


snipMate插件安装-1





snipMate插件安装-2





快捷键定义文件

snippets

文件(F) 编辑(E) 查看(V) 收藏(A) 工具(T) 帮助(H)

后退 搜索 文件夹

地址(D) C:\Program Files\Vim\vimfiles\snippets 转到

名称	大小	类型	修改日期	持续时间	尺寸
._. snippets	1 KB	SNIPPETS 文件	2009-7-13 11:19		
autoit. snippets	2 KB	SNIPPETS 文件	2009-7-13 11:19		
c. snippets	3 KB	SNIPPETS 文件	2009-7-13 11:22		
cpp. snippets	1 KB	SNIPPETS 文件	2009-7-13 11:19		
html. snippets	8 KB	SNIPPETS 文件	2012-3-23 18:27		
java. snippets	2 KB	SNIPPETS 文件	2009-7-13 11:19		
javascript. snippets	2 KB	SNIPPETS 文件	2009-7-13 11:19		
mako. snippets	1 KB	SNIPPETS 文件	2009-7-13 11:19		
objc. snippets	4 KB	SNIPPETS 文件	2009-7-13 11:22		
perl. snippets	2 KB	SNIPPETS 文件	2009-7-13 11:19		
php. snippets	10 KB	SNIPPETS 文件	2012-3-23 18:28		
python. snippets	2 KB	SNIPPETS 文件	2009-7-13 11:19		
ruby. snippets	9 KB	SNIPPETS 文件	2009-7-13 11:19		
sh. snippets	1 KB	SNIPPETS 文件	2009-7-13 11:19		
snippet. snippets	1 KB	SNIPPETS 文件	2009-7-13 11:19		
tcl. snippets	2 KB	SNIPPETS 文件	2009-7-13 11:19		
tex. snippets	2 KB	SNIPPETS 文件	2009-7-13 11:19		
vim. snippets	1 KB	SNIPPETS 文件	2009-7-13 11:19		
zsh. snippets	1 KB	SNIPPETS 文件	2009-7-13 11:19		

html快捷键文件

js快捷键文件

php快捷键文件

shell快捷键文件

选定 4 个对象 18.2 KB 我的电脑



snipMate html快捷键使用

```
html.snippets (C:\Program Files\vim\vimfiles\snippets) - GVIM
文件(F) 编辑(E) 工具(T) 语法(S) 缓冲区(B) 窗口(W) 帮助(H)
179 snippet label
180 +- 1 行: <label for="${2:$1}">${1:name}</label><input type="${3:text/submi
181 snippet link
182 +- 1 行: <link rel="${1:stylesheet}" href="${2:/css/master.css}" type="tex
183 snippet mailto
184 +- 1 行: <a href="mailto:${1:joe@example.com}?subject=${2:feedback}">${3:e
185 snippet meta
186     <meta name="${1:name}" content="${2:content}"`Close()`>${3}
187 snippet opt
188     <option value="${1:option}">${2:$1}</option>${3}
189 snippet optt
190     <option>${1:option}</option>${2}
191 snippet select
192 +- 3 行: <select name="${1:some_name}" id="${2:$1}">-----
195 snippet table
196 +- 4 行: <table border="${1:1}" width="700px">-----
200 snippet textarea
201 +- 1 行: <textarea name="${1:Name}" rows="${2:8}" cols="${3:40}">${4}</tex
202 snippet img
203     ${4}
204 snippet a
205     <a href="${1}">${2:example}</a>
206 snippet frame
207 +- 7 行: <frameset rows="80,*"> -----
207, 1 68%
```



gvim html代码测试

html+tab

```
<html>
```

```
</html>
```

head+tab

```
<head>
```

```
</head>
```

body+tab

```
<body>
```

```
</body>
```



snipMate js快捷键使用

```
javascript.snippets (C:\Program Files\Vim\vimfiles\snippets) - GVIM
文件(F) 编辑(E) 工具(T) 语法(S) 缓冲区(B) 窗口(W) 帮助(H)

1 # Prototype
2 snippet proto
3 +-- 4 行: ${1:class_name}.prototype.${2:method_name} =-----
7 # Function
8 snippet fun
9 +-- 3 行: function ${1:function_name} (${2:argument}) {-----
12 # Anonymous Function
13 snippet f
14     function(${1}) ${2};
15 # if
16 snippet if
17     if (${1:true}) ${2};
18 # if ... else
19 snippet ife
20 +-- 2 行: if (${1:true}) ${2}-----
22 # tertiary conditional
23 snippet t
24     ${1:/* condition */} ? ${2:a} : ${3:b}
25 # switch
26 snippet switch
27 +-- 8 行: switch(${1:expression}) {-----
35 # case
36 snippet case
37 +-- 4 行: case '${1:case}':-----

37, 9-1 顶端
```




gvim js代码测试

for+tab

```
for (var i = 0; i < Things.length; i++) {  
    Things[i]  
};
```

if+tab

```
if (true) { };
```

alert+tab

```
alert("example...");
```



snipMate php快捷键使用

```
php.snippets (C:\Program Files\vim\vimfiles\snippets) - GVIM
文件(F) 编辑(E) 工具(T) 语法(S) 缓冲区(B) 窗口(W) 帮助(H)

1 snippet php
2 +-- 2 行: <?php-----
4 snippet ec
5     echo "${1:string}"${2};
6 snippet inc
7     include '${1:file}';${2}
8 snippet incl
9     include_once '${1:file}';${2}
10 snippet req
11     require '${1:file}';${2}
12 snippet req1
13     require_once '${1:file}';${2}
14 # $GLOBALS['...']
15 snippet globals
16     $GLOBALS['${1:variable}']${2: = }${3:something}${4:;}${5}
17 snippet $_COOKIE['...']
18     $_COOKIE['${1:variable}']${2}
19 snippet $_ENV['...']
20     $_ENV['${1:variable}']${2}
21 snippet $_FILES['...']
22     $_FILES['${1:variable}']${2}
23 snippet $_GET['...']
24     $_GET['${1:variable}']${2}
25 snippet $_POST['...']
```



gvim php代码测试

php+tab

```
<?php
```

foreach+tab

```
foreach ($variable as $key) {  
    // code...  
}
```

case+tab

```
case 'value':  
    // code...  
    break;
```



代码编码问题

书写代码时，文本的编码更改：

```
index.html (~\桌面) - GVIM
文件(F) 编辑(E) 工具(T) 语法(S) 缓冲区(B) 窗口(W) 帮助(H)

1 <html>
2   <body>
3     <form action="" method="get" accept-charset="utf-8">
4       <p><input type="text" value="text &rarr;"></p>
5       <p><input type="password" value="password &rarr;"></p>
6       <p><input type="submit" value="Continue &rarr;"></p>
7     </form>
8     <font color="#ff0000">兄弟连</font>
9     <font color="#ff0000">兄弟连</font>
10    <font color="#ff0000">兄弟连</font>
11  </body>
12 </html>

~
~
~

: set fileencoding=utf-8 或 set fileencoding=gbk
```

用set fileencoding=gbk即中文编码



本章总结

- 1.gvim配置文件
- 2.gvim自动缩进
- 3.gvim页面编码
- 4.gvim 函数补全功能
- 5.gvim snipMate快捷键制作



谢谢大家！



姓名:李强强

QQ:447096447

邮箱:liqiangqiang@lampbrother.net