

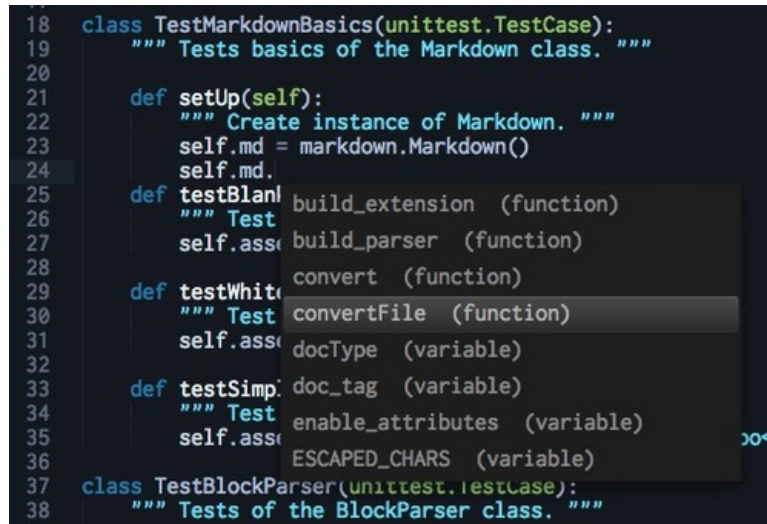
常用的sublime text 3插件（很爽哦） - Peter_zhou - 博客园(1)

个人比较懒，平时喜欢用webstorm，但是因为webstorm打开实在太慢了，并且太看设备，所以本人编辑简单的文件依然会选择使用sublime，虽然网上有很多关于此类插件的分享了，但是感觉都是片段，没有重点细致整理。

今天休息，在这简单的分享下自己常用插件

- All Autocomplete

Sublime Text 默认的 Autocomplete 功能只考虑当前的文件，而 AllAutocomplete 插件会搜索所有打开的文件来寻找匹配的提示词。



- SublimeCodeIntel

一个全功能的 Sublime Text 代码自动完成引擎，本人做过对比，但是如果和webstorm的自动寻找还是稍逊一筹，不过对于大部分人来说够用了，能很方便跳到你想要的方法

支持的语言挺多的（JavaScript, Mason, XBL, XUL, RHTML, SCSS, Python, HTML, Ruby, Python3, XML, Sass, XSLT, Django, HTML5, Perl, CSS, Twig, Less, Smarty, Node.js, Tcl, TemplateToolkit, PHP.）

- CTags

实在方法跳转，跳转到你方法

之后在win7下或者linux下安装ctags软件

1. 打开ctags插件包的use-setting配置"command": "d:/IDE/ctags58/ctags.exe"这个路径是下载ctags的安装路径
2. 这个插件能跨文件跳转，跳转到指定函数声明的地方(ctr+alt+左键)。使用package control 搜索ctags 进行安装（安装ctags 插件就可以了，还有一个 CTags for PHP 插件没什么用），注意安装好插件后要需要安装ctags命令。window 下载 ctags.exe <http://vdisk.weibo.com/s/7QZd7>。将ctags.exe文件放在一个环境变量能访问到的地方。打开cmd，输入ctags，如果有这个命令，证明成功了。ubuntu下安装运行命令：sudo apt-get install exuberant-ctags。然后在sublime项目文件夹右键，会出现Ctag:Rebuild Tags 的菜单。点击它，然后会生成.tags的文件 然后在你代码中，光标放在某个函数上，点击 就可以跳转到函数声明的地方。

- AutoFileName

快速帮助你在文件中写路径整体来说还不错

- Autoprefixer

这个插件主要应用css的浏览器兼容书写，自动分析你的css文件，解析出新的css文件，可以配置你要兼容的浏览器，不过这个插件要在之前安装node.js

```
body {
  transition: all 1s ease-out;
}

p {
  transform: skewx(10deg) translateX(150px);
}
```

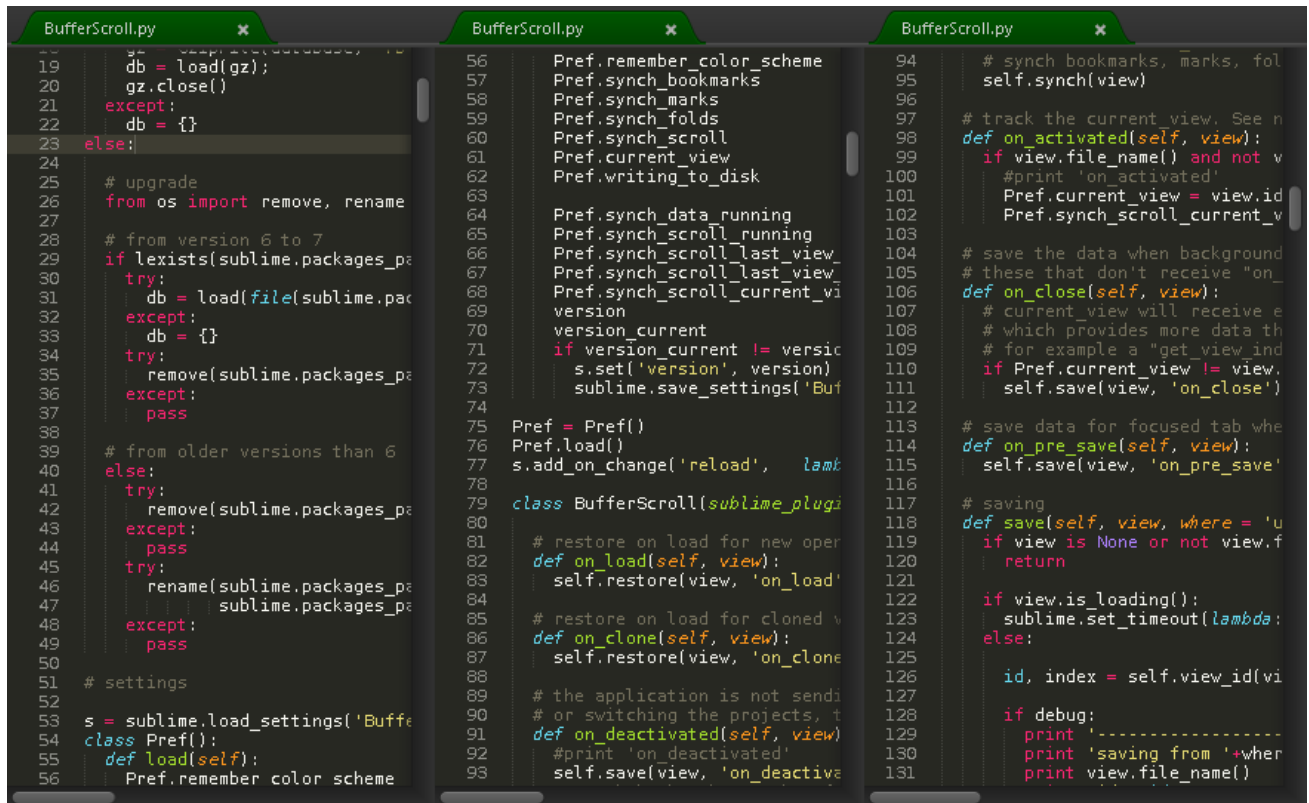
- BracketHighlighter

配置文件的高亮设置，让你的代码有不同的颜色区分该插件提供配对标签，或大括号或字符引号的配对高亮显示，算是对系统高亮的加强吧。

```
1 <?php
2 function test_function($var1) {
3     $var2[1] = test();
4     $var2[2] = $var1;
5 }
6
7 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4
8 <html>
9 <head>
10 </head>
11 <body>
12 <div>Test HTML</div>
13 </body>
14 </html>
```

- BufferScroll

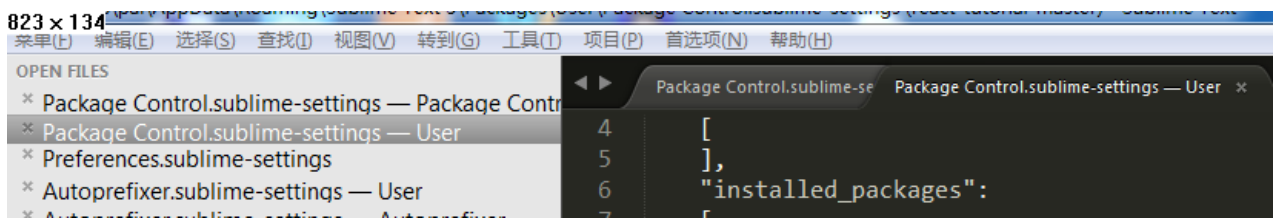
你可以轻松书写一个文件多个位置了



```
BufferScroll.py
19 db = load(gz);
20 gz.close()
21 except:
22     db = {}
23 else:
24
25 # upgrade
26 from os import remove, rename
27
28 # from version 6 to 7
29 if lexists(sublime.packages_path):
30     try:
31         db = load(file(sublime.packages_path))
32     except:
33         db = {}
34     try:
35         remove(sublime.packages_path)
36     except:
37         pass
38
39 # from older versions than 6
40 else:
41     try:
42         remove(sublime.packages_path)
43     except:
44         pass
45     try:
46         rename(sublime.packages_path, sublime.packages_path + '.old')
47     except:
48         pass
49
50 # settings
51 s = sublime.load_settings('BufferScroll.sublime-settings')
52 class Pref():
53     def load(self):
54         Pref.remember_color_scheme
55
56 Pref.remember_color_scheme
57 Pref.synch_bookmarks
58 Pref.synch_marks
59 Pref.synch_folds
60 Pref.synch_scroll
61 Pref.current_view
62 Pref.writing_to_disk
63
64 Pref.synch_data_running
65 Pref.synch_scroll_running
66 Pref.synch_scroll_last_view
67 Pref.synch_scroll_last_view_id
68 Pref.synch_scroll_current_view
69 version
70 version_current
71 if version_current != version:
72     s.set('version', version)
73     sublime.save_settings('BufferScroll.sublime-settings')
74
75 Pref = Pref()
76 Pref.load()
77 s.add_on_change('reload', lambda: Pref.load())
78
79 class BufferScroll(sublime_plugin):
80
81     # restore on load for new operation
82     def on_load(self, view):
83         self.restore(view, 'on_load')
84
85     # restore on load for cloned view
86     def on_clone(self, view):
87         self.restore(view, 'on_clone')
88
89     # the application is not sending a message
90     # or switching the projects, then
91     def on_deactivated(self, view):
92         #print 'on_deactivated'
93         self.save(view, 'on_deactivated')
94
95 # synch bookmarks, marks, folds
96 self.synch(view)
97
98 # track the current view. See on_activated
99 def on_activated(self, view):
100     if view.file_name() and not view.is_loading():
101         #print 'on_activated'
102         Pref.current_view = view.id
103         Pref.synch_scroll_current_view = view
104
105 # save the data when background
106 # these that don't receive 'on_pre_save'
107 def on_close(self, view):
108     # current view will receive a message
109     # which provides more data than
110     # for example a 'get_view_index'
111     if Pref.current_view != view.id:
112         self.save(view, 'on_close')
113
114 # save data for focused tab when
115 def on_pre_save(self, view):
116     self.save(view, 'on_pre_save')
117
118 # saving
119 def save(self, view, where = 'user_data'):
120     if view is None or not view.file_name():
121         return
122
123     if view.is_loading():
124         sublime.set_timeout(lambda: self.save(view, where), 100)
125     else:
126         id, index = self.view_id(view)
127
128     if debug:
129         print '-----'
130         print 'saving from ' + where
131         print view.file_name()
```

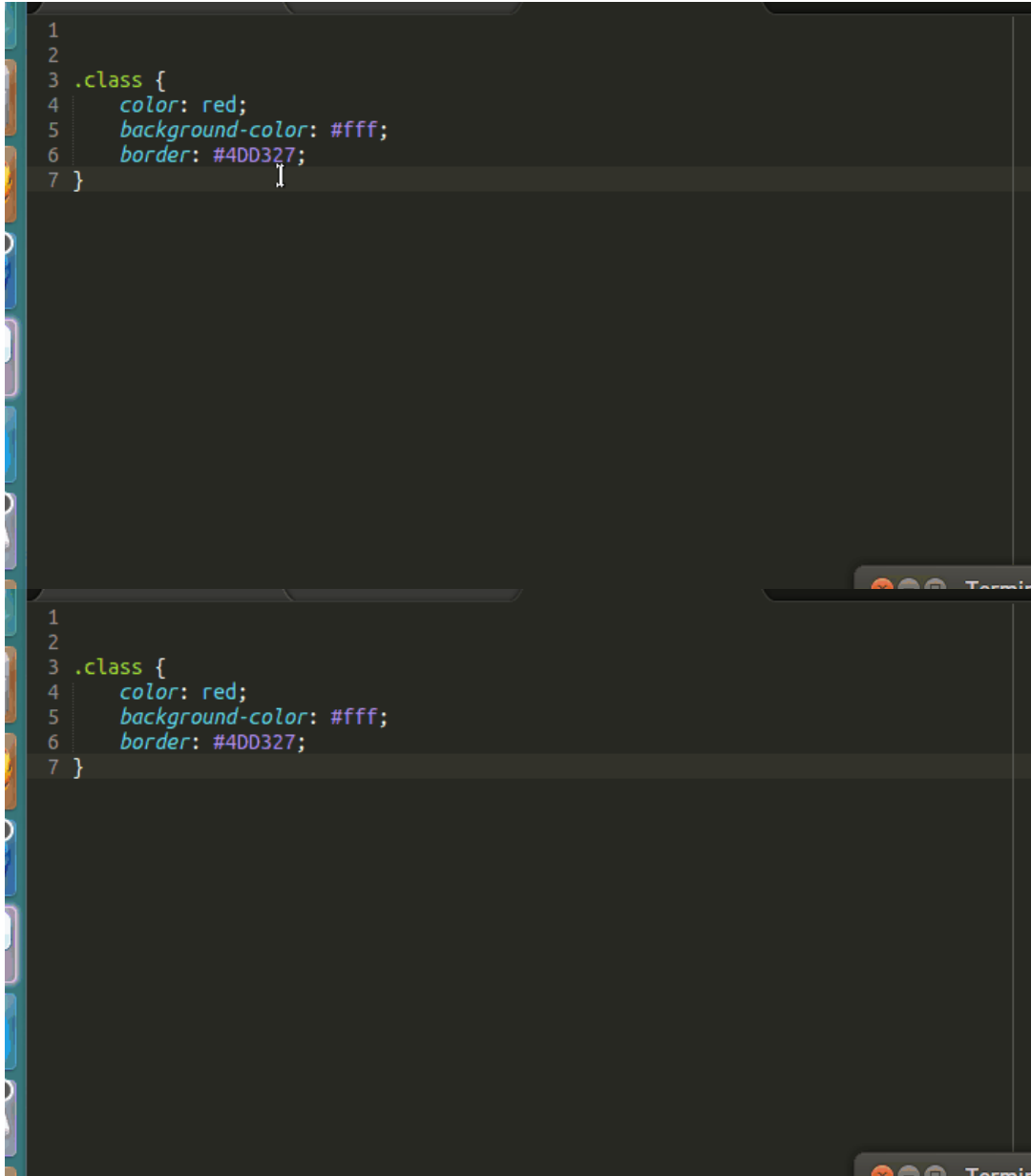
- ChineseLocalization

各国语言包，小日本的都有哦



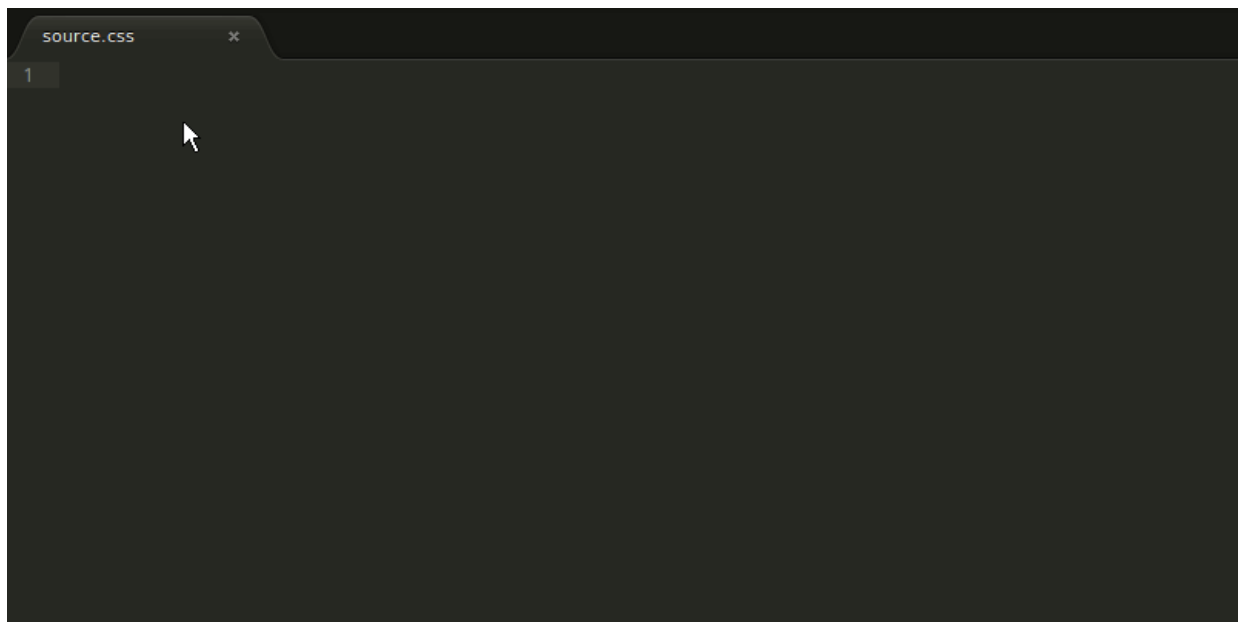
- Color Highlighter

颜色功能还是很爽的，找了好久



- CSS Comments

该有的都有，不该有的也有了



- CSS Format

css序列化插件，支持默认多种序列方案，还可以自己配置自己喜欢的

- CSS3

css3语言提示插件，本来不想写的，也不是什么特别的，但是可能会有人用到

- DocBlockr

DocBlocker 是在Sublime平台上开发一款自动补全代码插件，支持JavaScript (including ES6), PHP, ActionScript, Haxe, CoffeeScript, TypeScript, Java, Apex, Groovy, Objective C, C, C++ and Rust. 等众多语言



```
1 <?php
2
3 function foo(Array $arr,
4               MyClass $cls,
5               $str = "abc",
6               $i = 0,
7               $b = false) {}
8
9
10
```

- Emmet

Emmet的前身是大名鼎鼎的Zen coding，如果你从事Web前端开发的话，对该插件一定不会陌生。它使用仿CSS选择器的语法来生成代码，大大提高了HTML/CSS代码编写的速度

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.
html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ru">
<head>
  <title>Title</title>
  <meta http-equiv="Content-Type" content="text/html; charset=utf
  meta: c
</head>
<body>

</body>
</html>
```

1

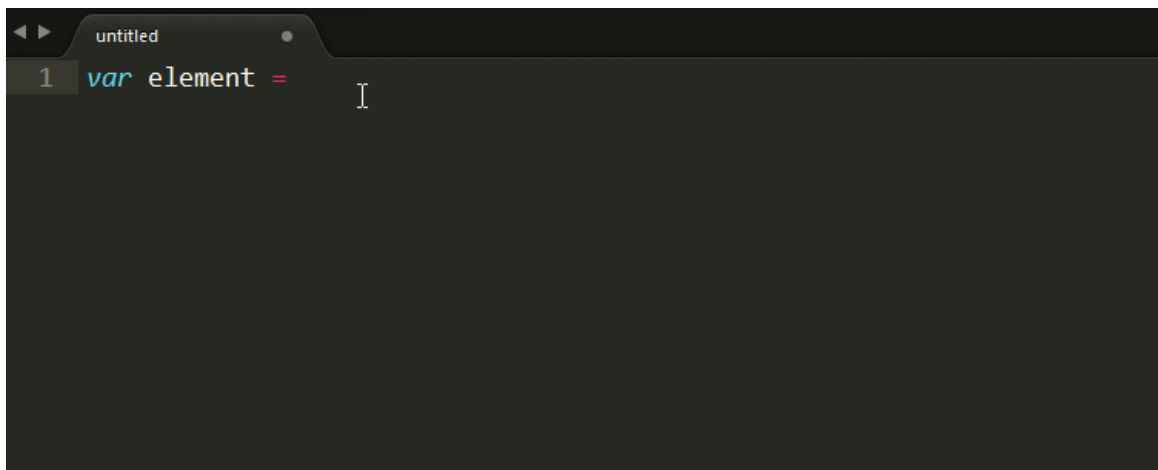
1

- HTML-CSS-JS Prettify

全能序列化

- JavaScript Completions

js最基本的api快查片段

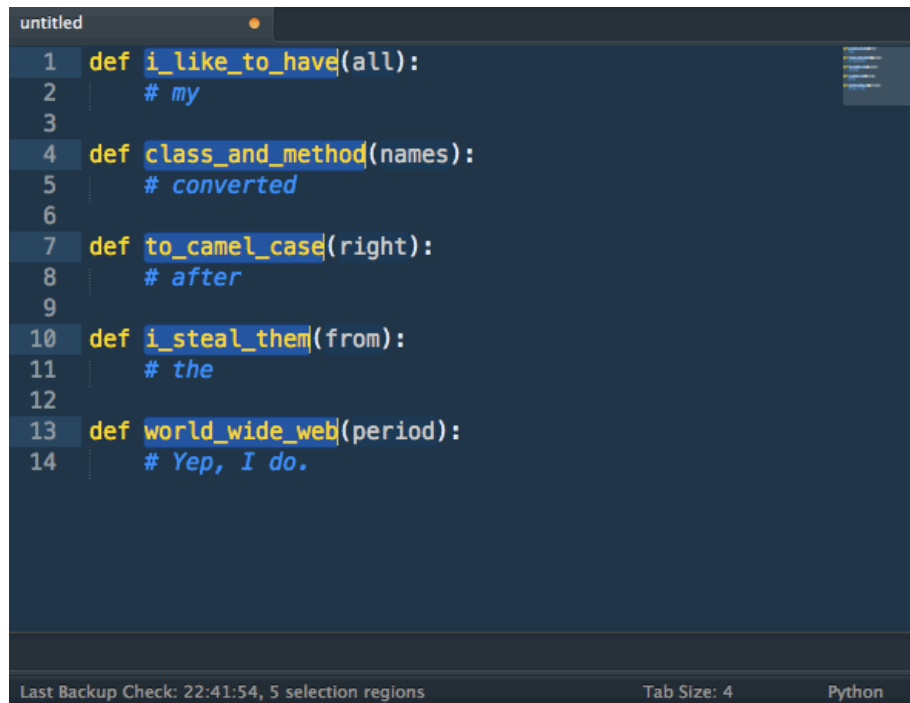


- JsFormat

js序列化，能排在下载插件前25位，好的话就不用说了

- Keymaps

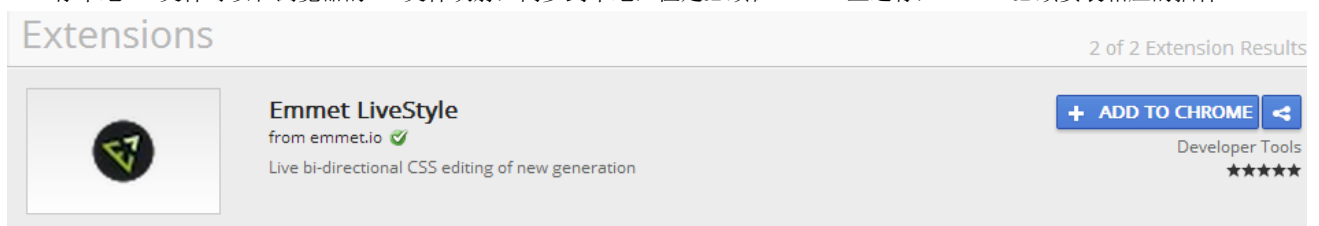
快速查找所有插件的快捷键



- LiveStyle

LiveStyle是Chrome中提高开发效率的一款CSS编辑器插件。利用LiveStyle和Sublime Text3编辑器结合可实现可视化开发，一次配置，简单易用！，并且最近支持less, scss

你本地css文件可以和浏览器的css文件映射，同步到本地，但是必须在chrome上运行，chrome必须安装相应的插件





[新闻](#) [网页](#) [贴吧](#) [知道](#) [音乐](#) [图片](#) [视频](#) [地图](#)

百度一下

[百科](#) [文库](#) [hao123](#) | [更多>>](#)

[把百度设为主页](#) [安装百度浏览器](#)

Elements Resources Network Sources Timeline Profiles Audits Console LiveStyle

☒ Enable LiveStyle for current page

Sublime Text 3

File mapping

Patch history

Associate web page files (on the left) with editor files (on the right). Open more CSS files in editor to make them appear in drop-down menus.

user CSS 1 E:\HiWiFicss\custom.css

Add file

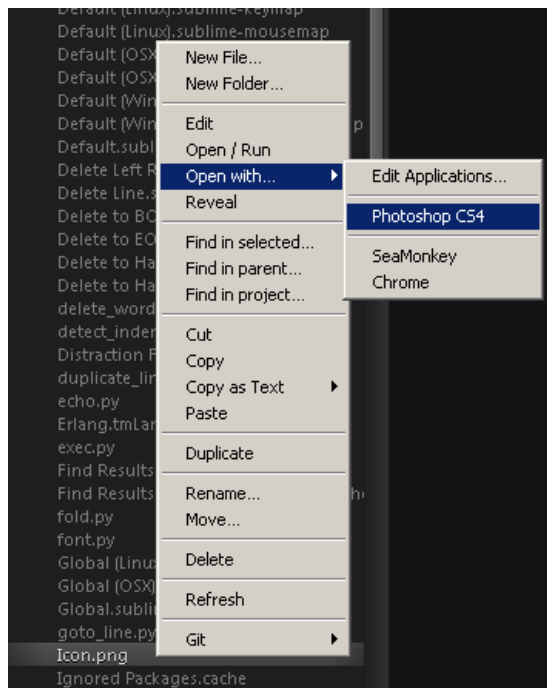
- Pretty JSON

JSON，一個輕量級的資料交換語言，目前許多網站AJAX request的回應結果都是JSON格式

```
1 for example here is dummy solr search output:
2
3 {"response":{
4   "docs":[{
5     "name":"Nik"
6   },{
7     "name":"Mike"
8   },{
9     "name":"Praful"
10  }],"start":0,"numFound":3
11 },"responseHeader":{
12   "QTime":0,"status":0
13 }}
14 }
```

- SideBarEnhancements

增强右键菜单文件操作功能



- SublimeLinter

代码校验插件，支持多种语言，这个是主插件，如果想检测特定的文件需要单独下载

- SublimeLinter-jshint

这个就是单独的插件，上面的一个分支

- SublimeTmpl

创建常用文件初始模板，必须html,css,js模板

- Tag

HTML/XML标签缩进、补全和校验

- Alignment

代码对齐

- PackageResourceViewer

通过这个特殊的插件，会给你查看和编辑SublimeText附带的不同的包带来很多方便。您也可以提取任何给定的包。这一行动将其复制到用户文件夹，以便您可以安全地对其进行编辑。

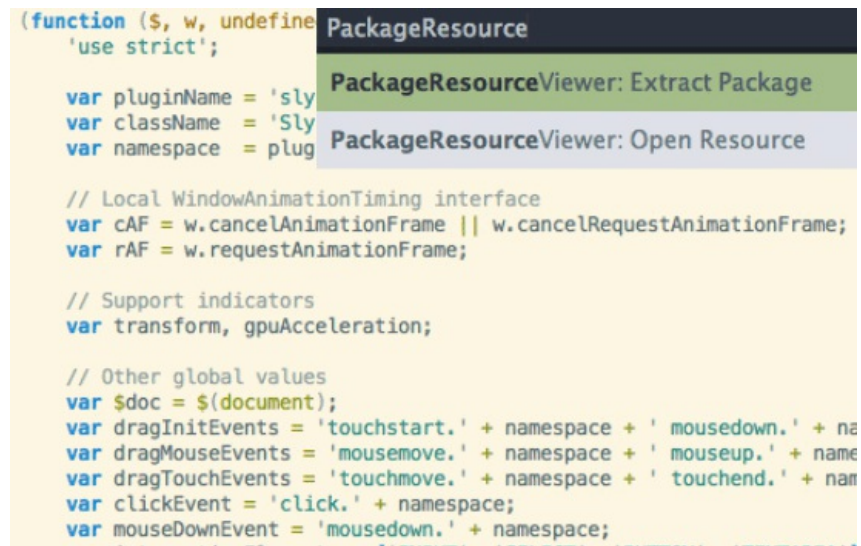
很多人苦恼不能修改左侧导航面板字体大小，用这个可以轻松办到

安装PackageResourceViewer 快捷键 $\text{⌘}(\text{command}) + \text{⇧}(\text{shift}) + \text{P}$ 打开 Command Palette 输入 Package

Control:Install 回车，等待加载package列表 搜索并安装 PackageResourceViewer 包

最后，使用PackageResourceViewer打开Theme文件进行编辑 快捷键 $\text{⌘}(\text{command}) + \text{⇧}(\text{shift}) + \text{P}$ 打开 Command Palette 输入 PackageResourceViewer: Open Resource 回车，打开包列表 选择 Theme - Default，再选择 Default.sublime-theme 搜索 sidebar_label，在 "class": "sidebar_label" 后边加一行: "font.size": 18，将字体大小设置为18，保存。好啦，大功告成！

如果觉得行间距太小，可以往上找下，有个class:"sidebartree"，调一下里边的rowpadding配置即可。



- Themr

sublime可以下载很多风格样式，用这个插件可以管理所有的风格

这些就是我们大部分要用到的，其它的我就不细说了，因为每个人不一样,比如说git，sass，svn这些你们可以自己查找插件的网址如下，你可以找到你喜欢的插件

<https://packagecontrol.io/browse>

最近出现sublime：3103版本好多没有激活码

今天在这补充下文章

```
-- BEGIN LICENSE --Michael BarnesSingle User LicenseEA7E-8213858A353C41 872A0D5C DF9B2950
AFF6F667C458EA6D 8EA3C286 98D1D650 131A97ABAA919AEC EF20E143 B361B1E7 4C8B7F04B085E65E 2F5F5360 8489D422
FB8FC1AA93F6323C FD7F7544 3F39C318 D95E6480FCCC7561 8A4A1741 68FA4223 ADCED07200C25BE DBBC4855 C4CFB774
C5EC138C0FEC1CEF D9DCECEC D3A5DAD1 01316C36--- END LICENSE ---
```