

目录

第 1 章 Sublime Text 中文教程全程指引	1
1.1 摘要	1
1.2 前言 (Prologue)	1
1.2.1 编辑器的选择 (Editor Choices)	1
1.2.2 个人背景 (Personal Background)	1
1.2.3 本文风格 (Writing Style)	1
1.2.4 编辑器 (Editor) vs 集成开发环境 (Integrated Development Environment, 下文简称 IDE)	2
1.3 安装 (Installation)	2
1.3.1 添加 Sublime Text 到环境变量	2
1.3.2 安装 Package Control	4
1.4 购买 (Purchase)	5
1.5 概览 (Tour)	6
1.5.1 基本概念 (Basic Concepts)	6
1.5.2 配置 (Settings)	7
1.6 编辑 (Editing)	7
1.6.1 基本编辑 (Basic Editing)	7
1.6.2 选择 (Selecting)	8
1.6.3 查找&替换 (Finding&Replacing)	9
1.6.4 跳转 (Jumping)	11
1.6.5 中文输入法的问题	13
1.6.6 文件夹 (Folders)	15
1.7 窗口&标签 (Windows & Tabs)	16
1.7.1 窗口 (Window)	16
1.7.2 标签 (Tab)	16
1.7.3 全屏 (Full Screen)	17
1.8 风格 (Styles)	18
1.8.1 一些设置 (Miscellaneous Settings)	20
1.8.2 主题 (Themes)	20
1.8.3 配色 (Color)	26
1.9 编码 (Coding)	27
1.9.1 良好实践 (Good Practices)	27
1.9.2 代码段 (Code Snippets)	27
1.9.3 格式化 (Formatting)	28
1.9.4 自动完成 (Auto Completion)	28
1.9.5 括号 (Brackets)	29
1.9.6 命令行 (Command Line)	29
1.10 其它 (Miscellaneous)	30
1.11 快捷键列表 (Shortcuts Cheatsheet)	30
1.11.1 通用 (General)	30
1.11.2 编辑 (Editing)	30
1.11.3 选择 (Selecting)	31

1.11.4	查找&替换 (Finding&Replacing)	31
1.11.5	跳转 (Jumping)	31
1.11.6	窗口 (Window)	31
1.11.7	屏幕 (Screen)	32
1.12	延伸阅读 (Further Reading)	32
1.12.1	书籍 (Books)	32
1.12.2	链接 (Links)	32
1.12.3	视频 (Videos)	32

第1章 Sublime Text 中文教程全程指引

1.1 摘要

本文系统全面的介绍了 Sublime Text，旨在成为最优秀的 Sublime Text 中文教程。

更新记录

2014/09/27: 完成初稿

2014/09/28:

更正打开控制台的快捷键为 **Ctrl + `**

更正全局替换的快捷键为 **Ctrl + Alt + Enter**

1.2 前言 (Prologue)

Sublime Text 是一款跨平台代码编辑器 (Code Editor)，从最初的 Sublime Text 1.0，到现在的 Sublime Text 3.0，Sublime Text 从一个不知名的编辑器演变到现在几乎是各平台首选的 GUI 编辑器。而这样优秀的编辑器却没有一个靠谱的中文教程，所以我试图通过本文弥补这个缺陷。

1.2.1 编辑器的选择 (Editor Choices)

从初学编程到现在，我用过的编辑器有 EditPlus、UltraEdit、Notepad++、Vim、TextMate 和 Sublime Text，如果让我从中推荐，我会毫不犹豫的推荐 Vim 和 Sublime Text，原因有下面几点：

1. **跨平台**：Vim 和 Sublime Text 均为跨平台编辑器（在 Linux、OS X 和 Windows 下均可使用）。作为一个程序员，切换系统是常有的事情，为了减少重复学习，使用一个跨平台的编辑器是很有必要的。

2. **可扩展**：Vim 和 Sublime Text 都是可扩展的 (Extensible)，并包含大量实用插件，我们可以通过安装自己领域的插件来成倍提高工作效率。

3. **互补**：Vim 和 Sublime Text 分别是命令行环境 (CLI) 和图形界面环境 (GUI) 下的最佳选择，同时使用两者会大大提高工作效率。

1.2.2 个人背景 (Personal Background)

我是一名非常典型的程序员：平时工作主要在 Linux 环境下使用 Java 和 Python，偶尔会用 HTML+CSS+JavaScript 编写网页；业余时会在 Windows 环境编写一些 C#程序（包括控制台程序 (Console Application) 和移动应用 (Mobile App)，也会玩一些非主流语言（比如 Haskell，ML 和 Ruby 等）以拓展见识。

所以这篇文章会以我的个人工作内容为主要使用场景 (Scenario)，尽管无法覆盖到所有的使用场景，但我认为依然可以覆盖到绝大部分，如果您认为我遗漏了什么内容，请在文章下面回复，我会尽量更新。

1.2.3 本文风格 (Writing Style)

受益于 K&R C 的写作风格，我倾向于以实际案例来讲解 Sublime Text 的功能，所以本文中的例子均源于我在实际开发时遇到的问题。

此外，把本文会使用大量动画 (GIF) 演示 Sublime Text 的编辑功能，因为我发现图片难以演示完整的编辑流程 (Workflow)，而视频又过于重量级。本文的 GIF 动画均使用

ScreenToGif 进行录制。

1.2.4 编辑器（Editor） vs 集成开发环境（Integrated Development Environment, 下文简称 IDE）

我经常看到一些程序员拿编辑器和 IDE 进行比较，诸如 Vim 比 Eclipse 强大或是 Visual Studio 太慢不如 Notepad++ 好使之类的讨论比比皆是，个人认为这些讨论没有意义，因为编辑器和 IDE 根本是面向两种不同使用场景的工具：

编辑器面向无语义的纯文本，不涉及领域逻辑，因此速度快体积小，适合编写单独的配置文件和动态语言脚本（Shell、Python 和 Ruby 等）。

IDE 面向有语义的代码，会涉及到大量领域逻辑，因此速度偏慢体积庞大，适合编写静态语言项目（Java、C++ 和 C# 等）。

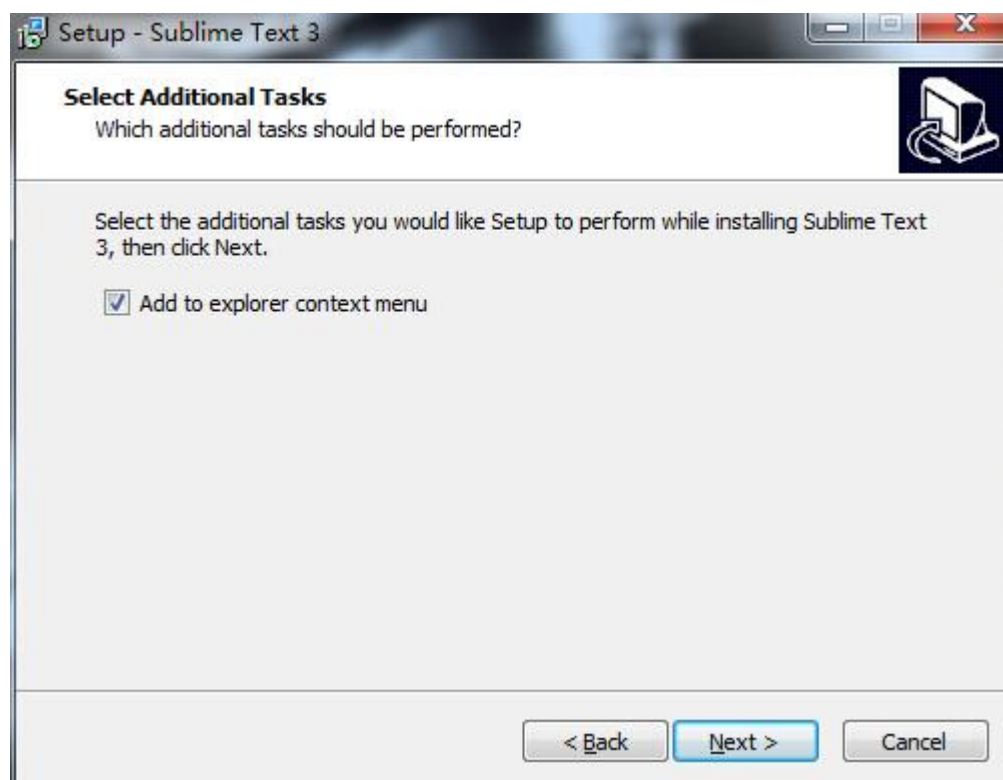
我认为应当使用正确的工具去做有价值的事情，并把效率最大化，所以我会用 Eclipse 编写 Java 项目，用 Vim 编写 Shell，用 Sublime Text 编写 JavaScript/HTML/Python，用 Visual Studio 编写 C#。

前言到此结束，下面进入正题。

1.3 安装（Installation）

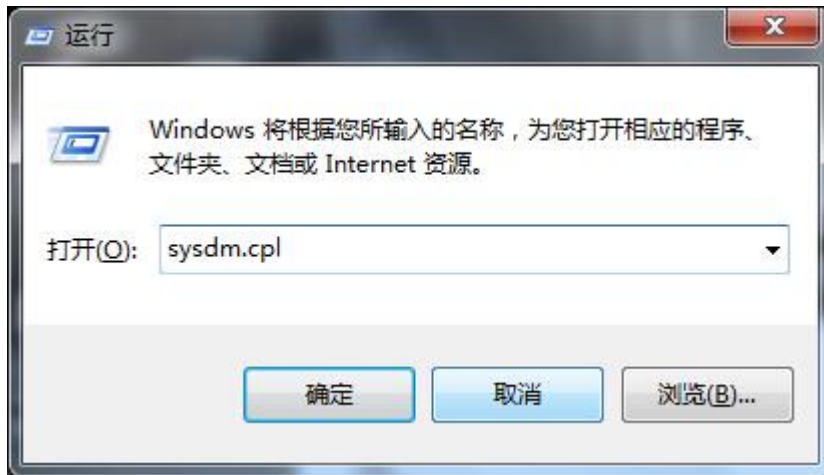
Sublime Text [官方网站](#)提供了 Sublime Text 各系统各版本的下载，目前 Sublime Text 的最新版本是 [Sublime Text 3](#)。这里以 Windows 版本的 Sublime Text 安装为例。

注意在安装时勾选 **Add to explorer context menu**，这样在右键单击文件时就可以直接使用 Sublime Text 打开。

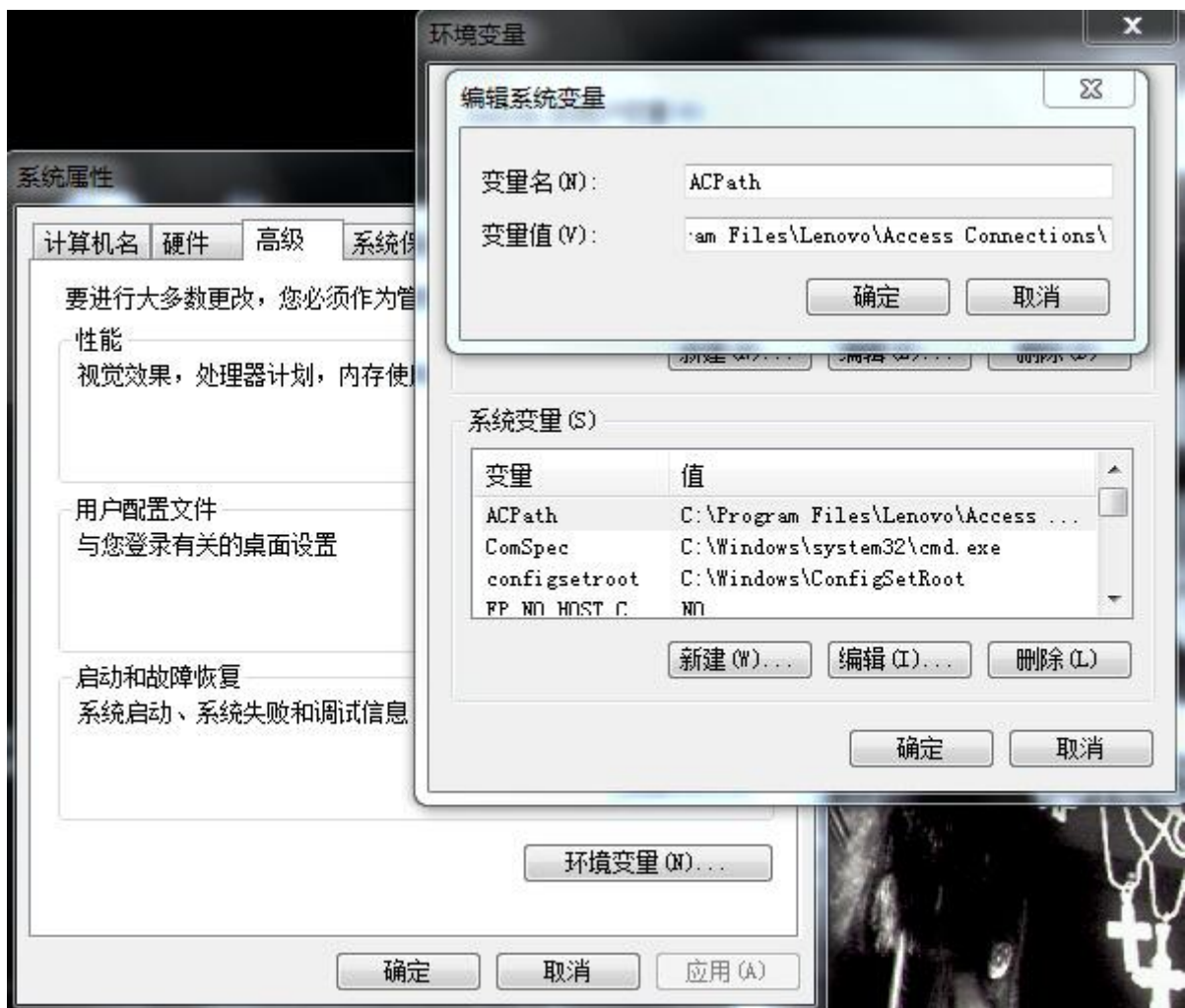


1.3.1 添加 Sublime Text 到环境变量

使用 Win + R 运行 `sysdm.cpl` 打开“系统属性”。



然后在“高级”选项卡里选择“环境变量”，编辑“Path”，增加 Sublime Text 的安装目录（例如 `D:\Program Files\Sublime Text 3`）。



接下来你就可以在命令行里面利用 `subl` 命令直接使用 Sublime Text 了：

- `subl file` :: 使用 Sublime Text 打开 file 文件
- `subl folder` :: 使用 Sublime Text 打开 folder 文件夹
- `subl .` :: 使用 Sublime Text 当前文件夹

1.3.2 安装 Package Control

前文提到 Sublime Text 支持大量插件，如何找到并管理这些插件就成了一个问题，Package Control 正是为了解决这个问题而出现的，利用它我们可以很方便的浏览、安装和卸载 Sublime Text 中的插件。

进入 Package Control 的[官网](#)，里面有详细的[安装教程](#)。Package Control 支持 Sublime Text 2 和 3，本文只给出 3 的安装流程：

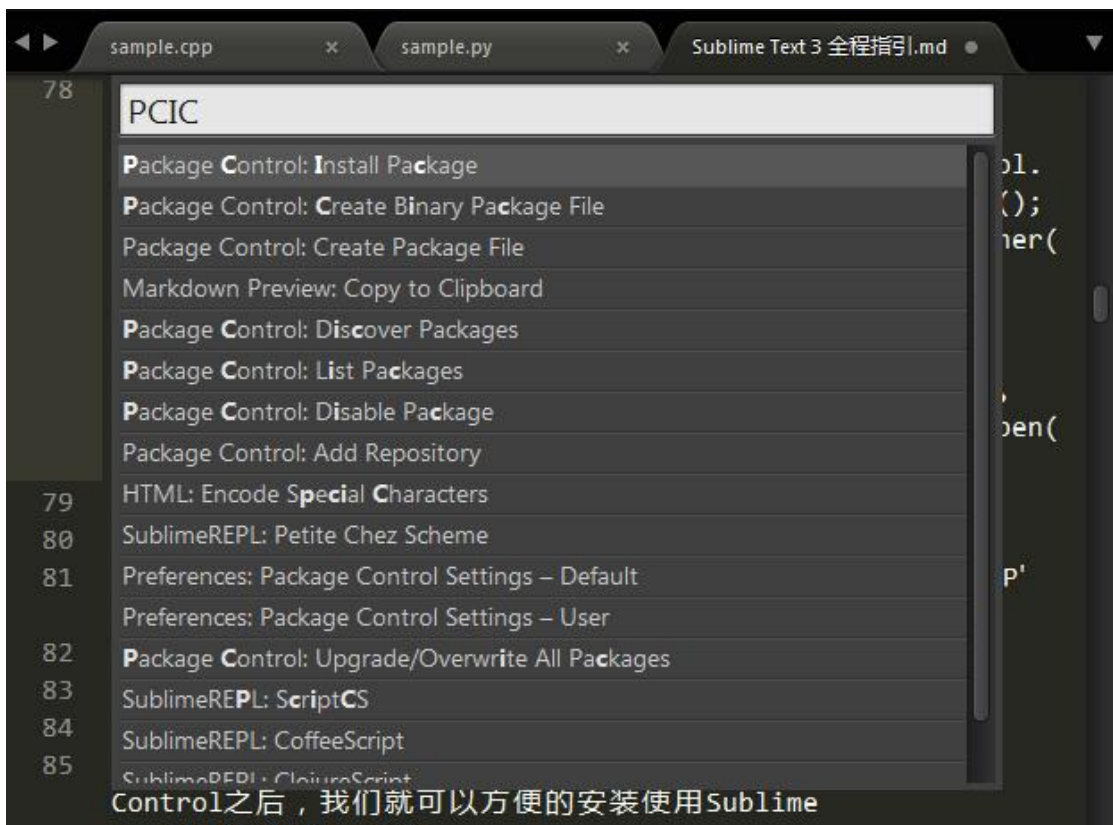
使用 **Ctrl + `** 打开 Sublime Text 控制台。

将下面的代码粘贴到控制台里：

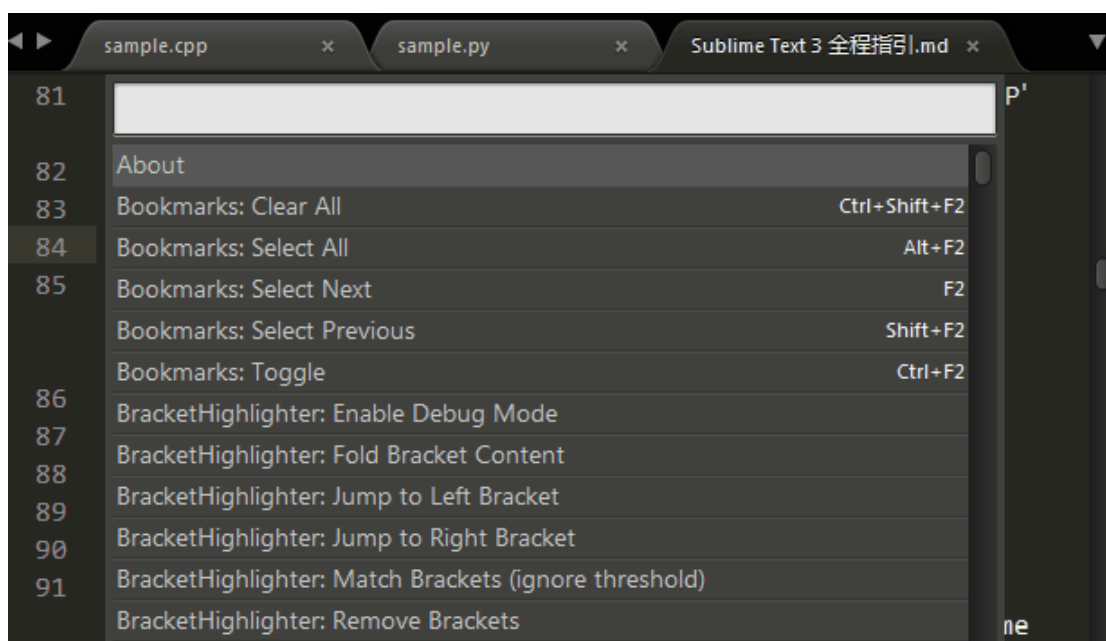
```
import urllib.request,os,hashlib; h =  
'7183a2d3e96f11eeadd761d777e62404' +  
'e330c659d4bb41d3bdf022e94cab3cd0'; pf = 'Package  
Control.sublime-package'; ipp = sublime.installed_packages_path();  
urllib.request.install_opener( urllib.request.build_opener( urllib.re  
quest.ProxyHandler()) ); by =  
urllib.request.urlopen( 'http://sublime.wbond.net/' + pf.replace(' ',  
'%20')).read(); dh = hashlib.sha256(by).hexdigest(); print('Error  
validating download (got %s instead of %s), please try manual install' %  
(dh, h)) if dh != h else open(os.path.join( ipp, pf), 'wb' ).write(by)
```

等待 Package Control 安装完成。之后使用 **Ctrl + Shift + P** 打开命令板，输入 **PC** 应出现 Package Control：

等待 Package Control 安装完成。之后使用 **Ctrl + Shift + P** 打开命令板，输入 **PC** 应出现 Package Control：



成功安装 Package Control 之后，我们就可以方便的安装使用 Sublime Text 的各种插件：



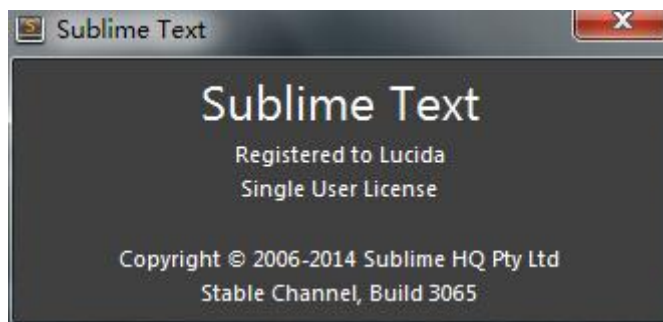
1.4 购买 (Purchase)

Sublime Text 是一个收费闭源软件，这在一定程度上成为了我支持 Sublime Text 的理由（我心中的软件靠谱程度：免费开源 << 免费闭源 < 收费开源 < 收费闭源）：[在这里](#)购买。

不过不购买 Sublime Text 也可以“正常”使用它，只是 Sublime Text 会时不时的弹出一个对话框提醒你购买，此外窗口处会有一个很屌丝很 low 逼的**(UNREGISTERED)**。（在高频操作下，一般 20 分钟提示一次，个人认为算是很厚道了）



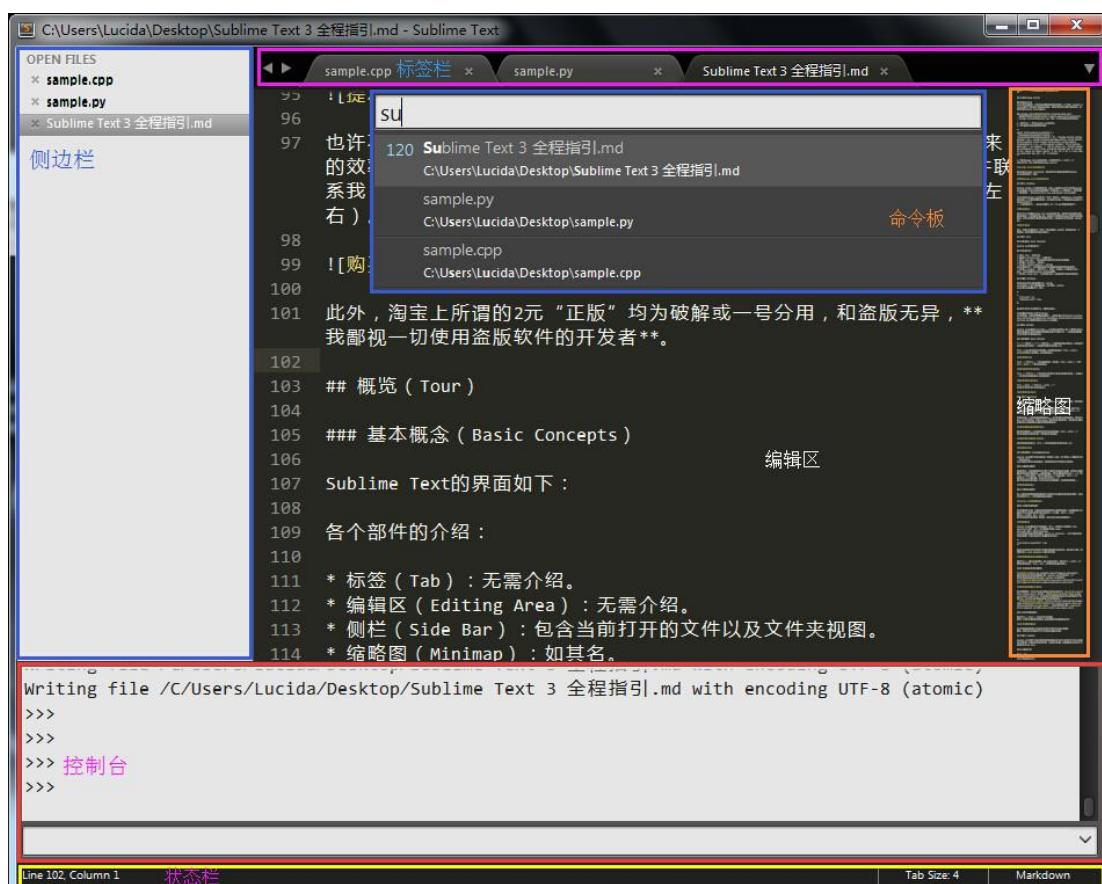
也许不少人会觉得 Sublime Text 70 刀的价格太贵，但相比它的功能和带来的效率提升，70 刀真的不值一提，如果你不方便使用 Paypal 付款可以邮件联系我，你支付宝给我打款然后我帮你付款，价格按当日汇率折算（450 元左右）。



1.5 概览 (Tour)

1.5.1 基本概念 (Basic Concepts)

Sublime Text 的界面如下：



- 标签 (Tab): 无需介绍。
- 编辑区 (Editing Area): 无需介绍。
- 侧栏 (Side Bar): 包含当前打开的文件以及文件夹视图。

- 缩略图 (Minimap): 如其名。
- 命令板 (Command Palette): Sublime Text 的操作中心, 它使得我们基本可以脱离鼠标和菜单栏进行操作。
- 控制台 (Console): 使用 **Ctrl + `** 调出, 它既是一个标准的 Python REPL, 也可以直接对 Sublime Text 进行配置。
- 状态栏 (Status Bar): 显示当前行号、当前语言和 Tab 格式等信息。

1.5.2 配置 (Settings)

与其他 GUI 环境下的编辑器不同, Sublime Text 并没有一个专门的配置界面, 与之相反, Sublime Text 使用 JSON 配置文件, 例如:

```
{  
  "font_size": 12,  
  "highlight_line": true,  
}
```

会将默认字体大小调整为 12, 并高亮当前行。

JSON 配置文件的引入简化了 Sublime Text 的界面, 但也使得配置变的复杂, 一般我会到[这里](#)查看可用的 Sublime Text 配置。

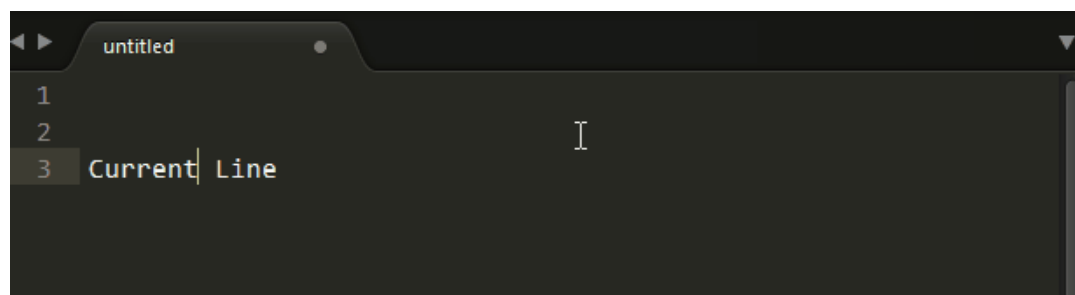
1.6 编辑 (Editing)

Sublime Text 的编辑十分人性化——它不像 Vim 那样反人类 (尽管我也用 Vim 但我还是要说 Vim 的快捷键设定绝壁连代谢产物都不如), 少量的快捷键就可以完成绝大多数编辑任务。

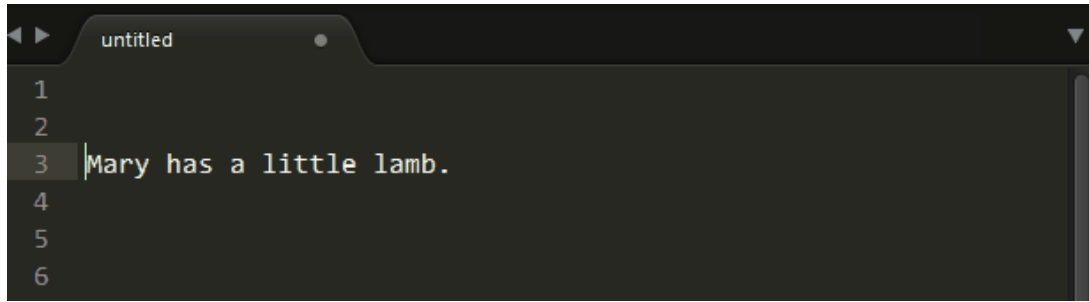
1.6.1 基本编辑 (Basic Editing)

↑ ↓ ← → 就是 **↑ ↓ ← →**, 不是 **KJHL**, (没错我就是在吐槽 Vim, 尼玛设成 **WSAD** 也比这个强啊), 粘贴剪切复制均和系统一致。

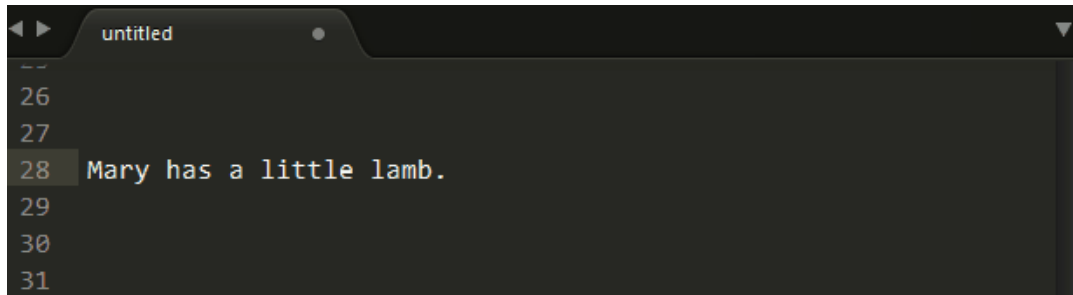
Ctrl + Enter 在当前行下面新增一行然后跳至该行; **Ctrl + Shift + Enter** 在当前行上面增加一行并跳至该行。



Ctrl + ←/→ 进行逐词移动, 相应的, **Ctrl + Shift + ←/→** 进行逐词选择。



Ctrl + ↑/↓ 移动当前显示区域，**Ctrl + Shift + ↑/↓** 移动当前行。

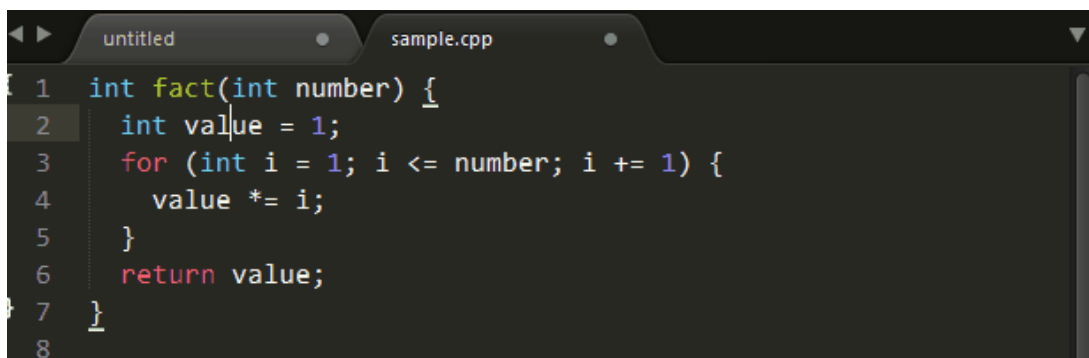


1.6.2 选择 (Selecting)

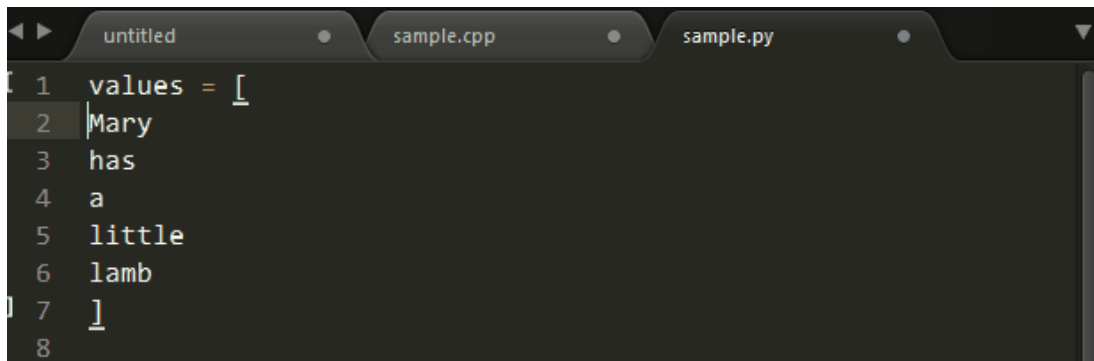
Sublime Text 的一大亮点是支持多重选择——同时选择多个区域，然后同时进行编辑。

Ctrl + D 选择当前光标所在的词并高亮该词所有出现的位置，再次 **Ctrl + D** 选择该词出现的下一个位置，在多重选词的过程中，使用 **Ctrl + K** 进行跳过，使用 **Ctrl + U** 进行回退，使用 **Esc** 退出多重编辑。

多重选词的一大应用场景就是重命名——从而使得代码更加整洁。尽管 Sublime Text 无法像 IDE（例如 Eclipse）那样进行自动重命名，但我们可以通过多重选词+多重编辑进行直观且便捷的重命名：



有时我们需要对一片区域的所有行进行同时编辑，**Ctrl + Shift + L** 可以将当前选中区域打散，然后进行同时编辑：



```
1 values = [  
2 Mary  
3 has  
4 a  
5 little  
6 lamb  
7 ]  
8
```

有打散自然就有合并，**Ctrl + J**可以把当前选中区域合并为一行：



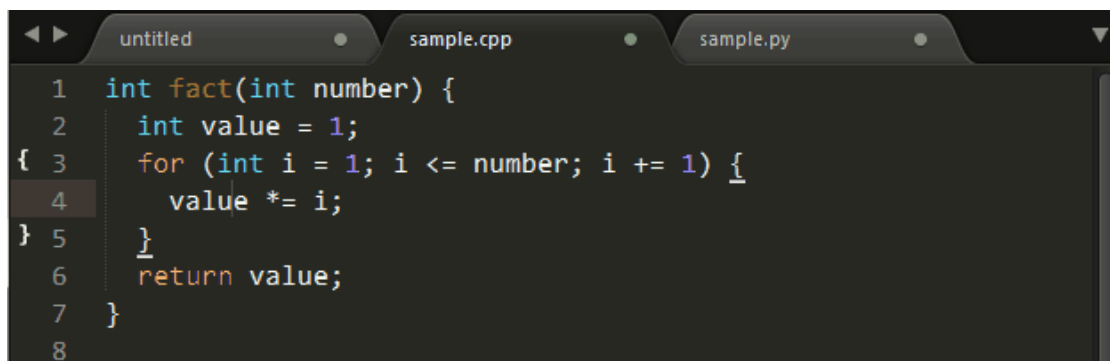
```
1 values = [  
2     'Mary',  
3     'has',  
4     'a',  
5     'little',  
6     'lamb'  
7 ]  
8
```

1.6.3 查找&替换 (Finding&Replacing)

Sublime Text 提供了强大的查找（和替换）功能，为了提供一个清晰的介绍，我将 Sublime Text 的查找功能分为**快速查找**、**标准查找**和**多文件查找**三种类型。

1. 快速查找&替换

多数情况下，我们需要查找文中某个关键字出现的其它位置，这时并不需要重新将该关键字重新输入一遍然后搜索，我们只需要使用 **Shift + ←/→**或 **Ctrl + D** 选中关键字，然后 **F3** 跳到其下一个出现位置，**Shift + F3** 跳到其上一个出现位置，此外还可以用 **Alt + F3** 选中其出现的所有位置（之后可以进行多重编辑，也就是快速替换）。



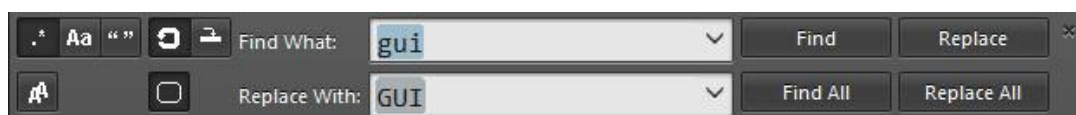
```
1 int fact(int number) {  
2     int value = 1;  
3     for (int i = 1; i <= number; i += 1) {  
4         value *= i;  
5     }  
6     return value;  
7 }  
8
```

2. 标准查找&替换

另一种常见的使用场景是搜索某个已知但不在当前显示区域的关键字，这时可以使用 **Ctrl + F** 调出搜索框进行搜索：



以及使用 **Ctrl + H** 进行替换：



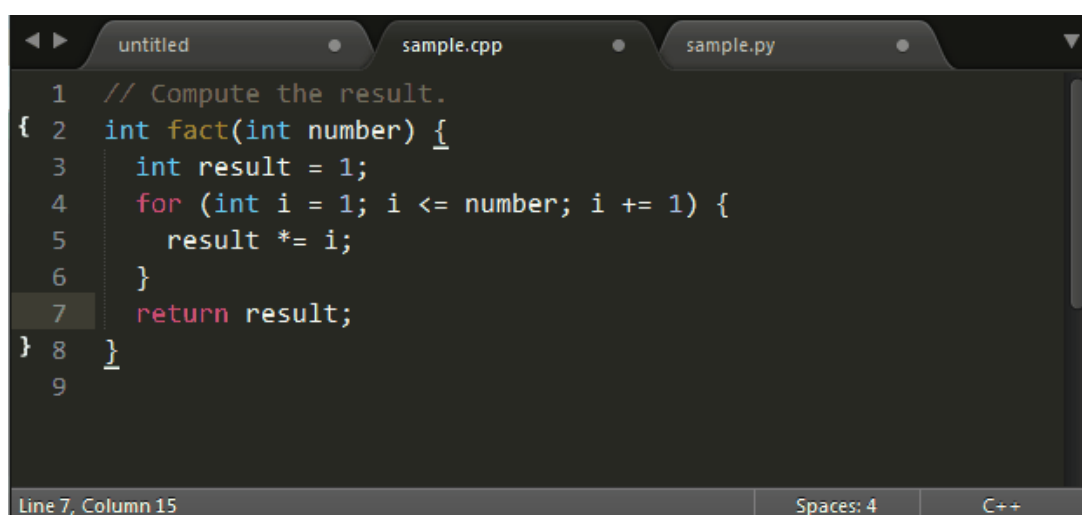
关键字查找&替换

对于普通用户来说，常规的关键字搜索就可以满足其需求：在搜索框输入关键字后 **Enter** 跳至关键字当前光标的下一个位置，**Shift + Enter** 跳至上一个位置，**Alt + Enter** 选中其出现的所有位置（同样的，接下来可以进行快速替换）。

Sublime Text 的查找有不同的模式：**Alt + C** 切换大小写敏感（Case-sensitive）模式，**Alt + W** 切换整字匹配（Whole matching）模式，除此之外 Sublime Text 还支持在选中范围内搜索（Search in selection），这个功能没有对应的快捷键，但可以通过以下配置项自动开启。

```
"auto_find_in_selection": true
```

这样之后在选中文本的状态下范围内搜索就会自动开启，配合这个功能，局部重命名（Local Renaming）变的非常方便：



使用 **Ctrl + H** 进行标准替换，输入替换内容后，使用 **Ctrl + Shift + H** 替换当前关键字，**Ctrl + Alt + Enter** 替换所有匹配关键字。

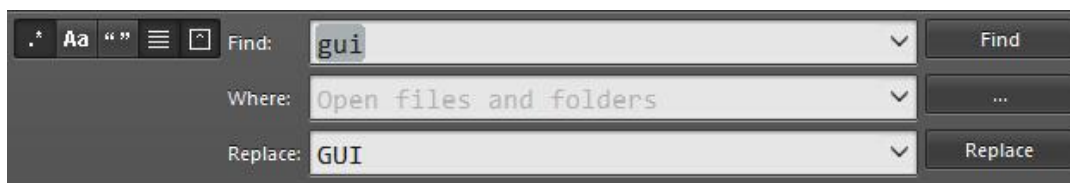
正则表达式查找&替换

正则表达式是非常强大的文本查找&替换工具，Sublime Text 中使用 **Alt + R** 切换正则匹配模式的开启/关闭。Sublime Text 的使用 Boost 里的 Perl 正则表达式风格。

出于篇幅原因，本文不会对正则表达式进行详细介绍，Mastering Regex（中译本：精通正则表达式）对正则表达式的原理和各语言下的使用进行了详细介绍。此外网上有大量正则表达式的优秀教程（“正则表达式 30 分钟入门教程”和 MSDN 正则表达式教程），以及在线测试工具（regexpal 和 regexer）。

3. 多文件搜索&替换

使用 **Ctrl + Shift + F** 开启多文件搜索&替换（注意此快捷键和搜狗输入法的简繁切换快捷键有冲突）：



多文件搜索&替换默认在当前打开的文件和文件夹进行搜索/替换，我们也可以指定文件/文件夹进行搜索/替换。

1.6.4 跳转（Jumping）

Sublime Text 提供了强大的跳转功能使得我们可以在不同的文件/方法/函数中无缝切换。就我的使用经验而言，目前还没有哪一款编辑器可以在这个方面超越 Sublime Text。

1. 跳转到文件

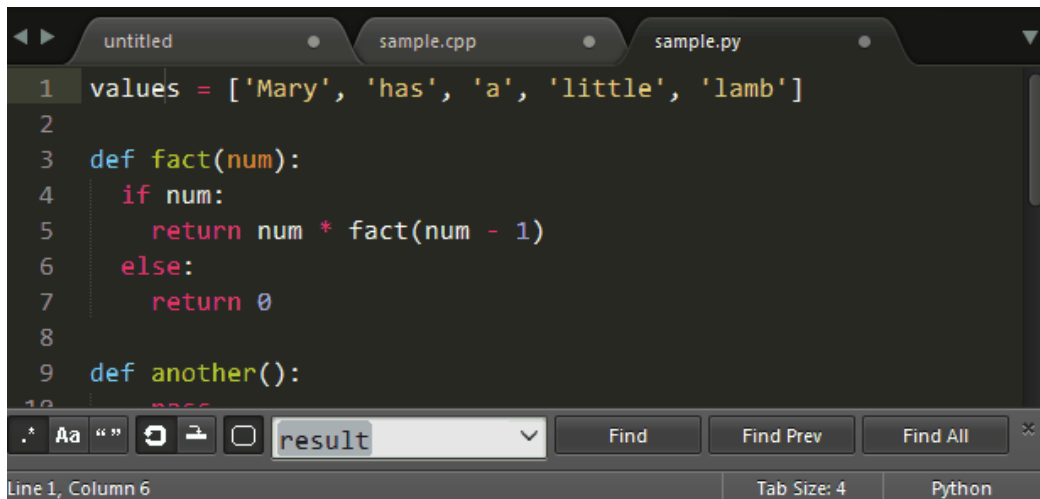
Ctrl + P 会列出当前打开的文件（或者是当前文件夹的文件），输入文件名然后 **Enter** 跳转至该文件。

需要注意的是，Sublime Text 使用模糊字符串匹配（Fuzzy String Matching），这也就意味着你可以通过文件名的前缀、首字母或是某部分进行匹配：例如，**EIS**、**Eclip** 和 **Stupid** 都可以匹配 **EclipseIsStupid.java**。



2. 跳转到符号

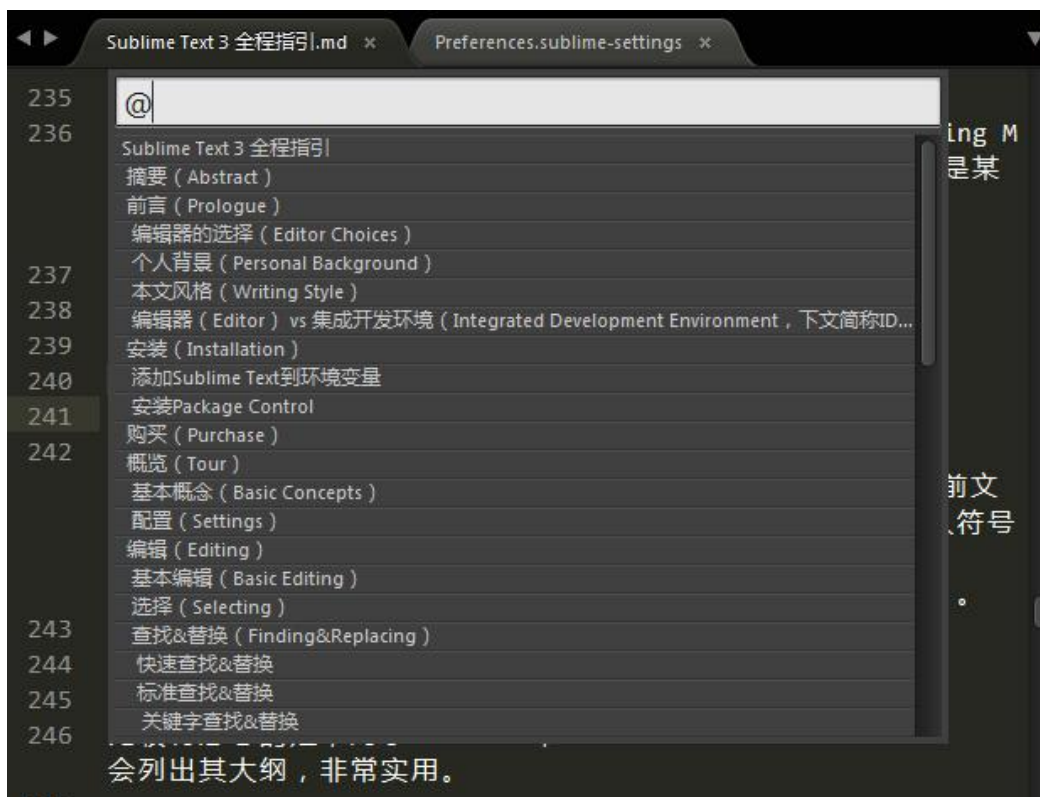
尽管是一个文本编辑器，Sublime Text 能够对代码符号进行一定程度的索引。**Ctrl + R** 会列出当前文件中的符号（例如类名和函数名，但无法深入到变量名），输入符号名称 **Enter** 即可以跳转到该处。此外，还可以使用 **F12** 快速跳转到当前光标所在符号的定义处（Jump to Definition）。



```
1 values = ['Mary', 'has', 'a', 'little', 'lamb']
2
3 def fact(num):
4     if num:
5         return num * fact(num - 1)
6     else:
7         return 0
8
9 def another():
10     pass
```

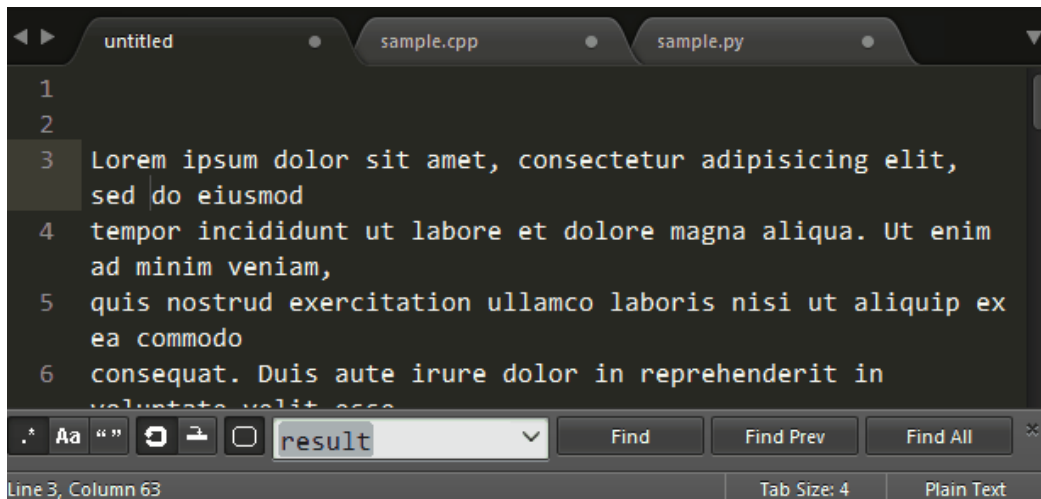
Line 1, Column 6 Tab Size: 4 Python

比较有意思的是，对于 Markdown，**Ctrl + R** 会列出其大纲，非常实用。



3. 跳转到某行

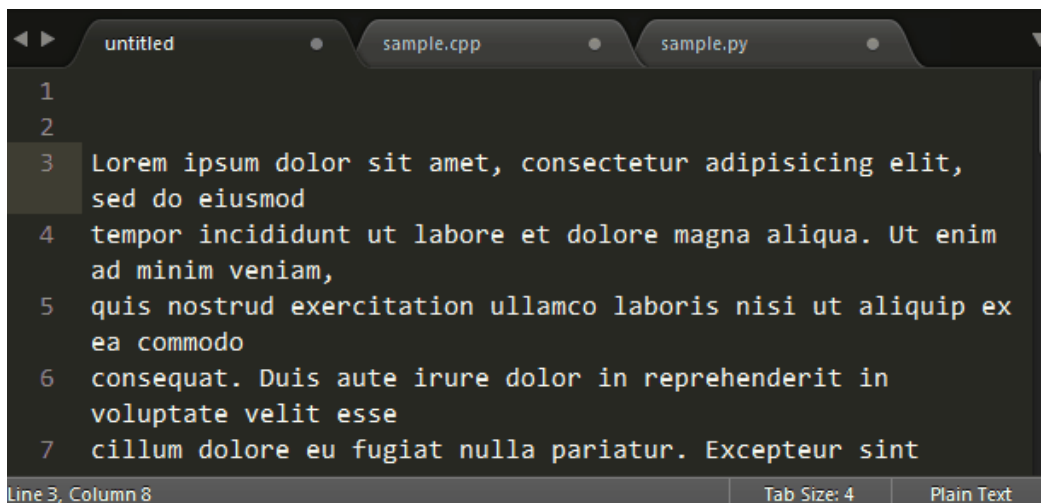
Ctrl + G 然后输入行号以跳转到指定行：



4. 组合跳转

在 **Ctrl + P** 匹配到文件后，我们可以进行后续输入以跳转到更精确的位置：

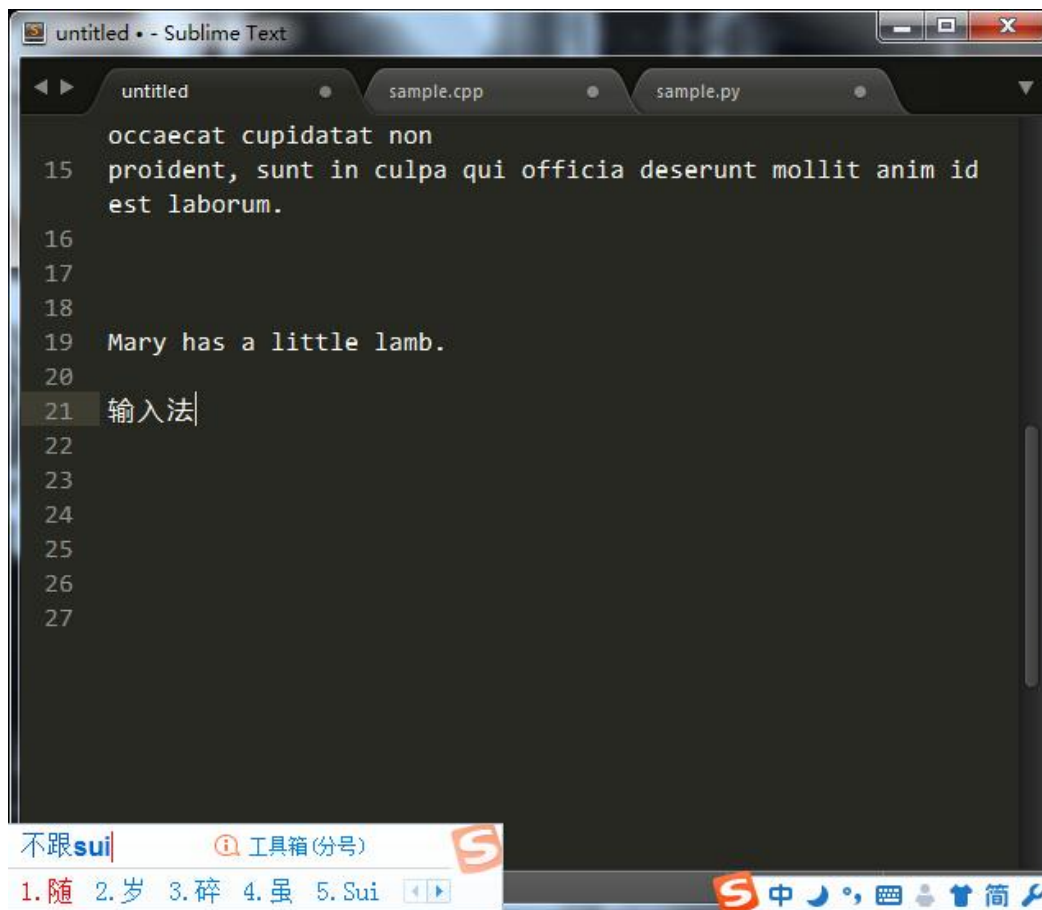
- **@** 符号跳转：输入 **@symbol** 跳转到 **symbol** 符号所在的位置
- **#** 关键字跳转：输入 **#keyword** 跳转到 **keyword** 所在的位置
- **:** 行号跳转：输入 **:12** 跳转到文件的第 12 行。



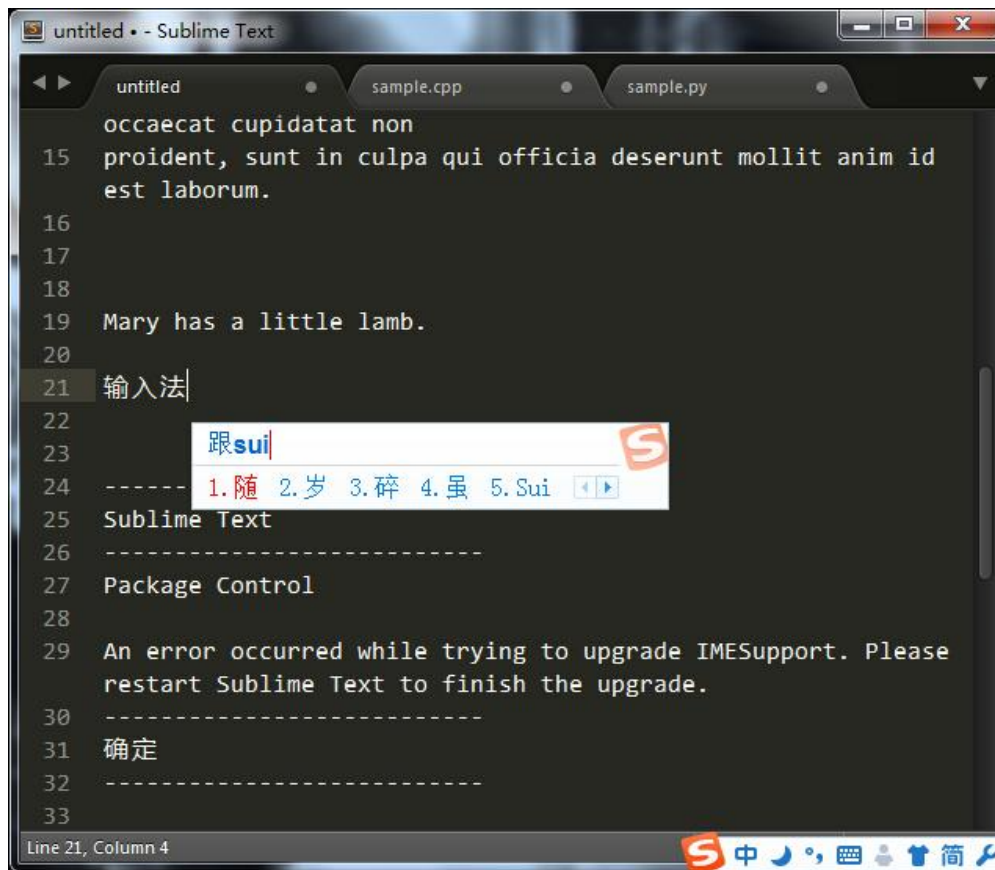
所以 Sublime Text 把 **Ctrl + P** 称之为“**Go To Anything**”，这个功能如此好用，以至于我认为没有其它编辑器能够超越它。

1.6.5 中文输入法的问题

从 Sublime Text 的初版（1.0）到现在（3.0 3065），中文输入法（包括日文输入法）都有一个问题：输入框不跟随。

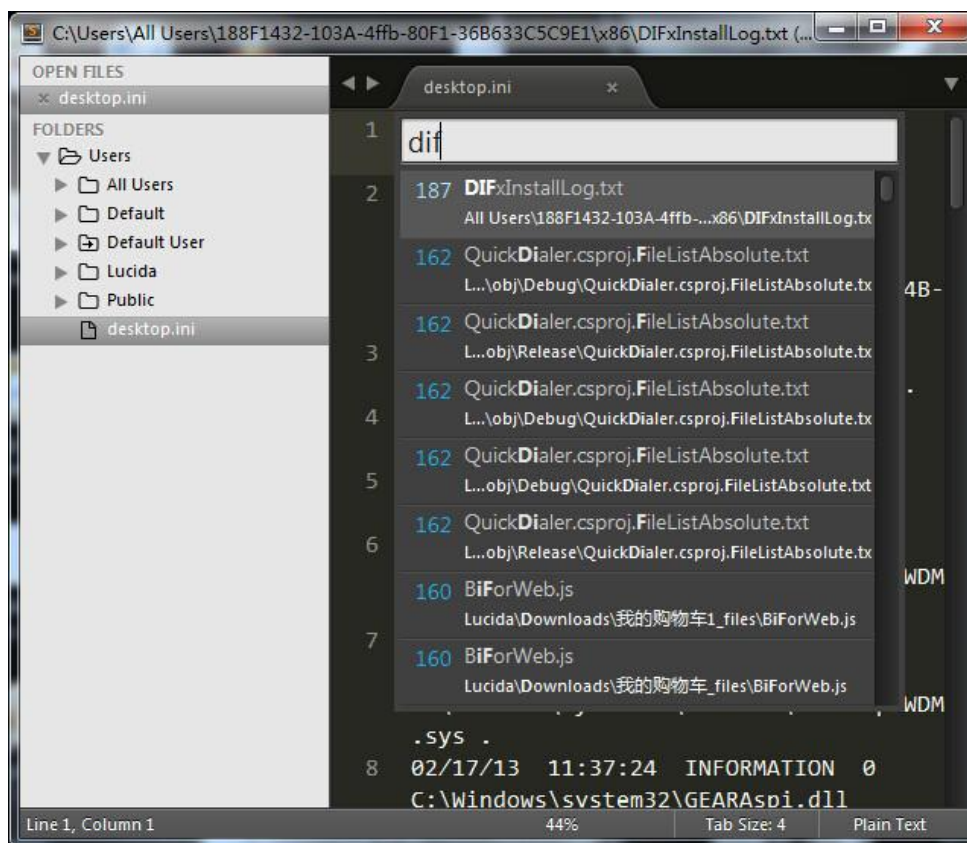


目前官方还没有修复这个 bug, 解决方法是安装 [IMESupport](#) 插件, 之后重启 Sublime Text 问题就解决了。



1.6.6 文件夹（Folders）

Sublime Text 支持以文件夹做为单位进行编辑，这在编辑一个文件夹下的代码时尤其有用。在 **File** 下 **Open Folder**:



你会发现右边多了一个侧栏，这个侧栏列出了当前打开的文件和文件夹的文件，使用 **Ctrl + K**, **Ctrl + B** 显示或隐藏侧栏，使用 **Ctrl + P** 快速跳转到文件夹里的文件。

1.7 窗口&标签 (Windows & Tabs)

Sublime Text 是一个多窗口多标签编辑器：我们既可以开多个 Sublime Text 窗口，也可以在一个 Sublime Text 窗口内开多个标签。

1.7.1 窗口 (Window)

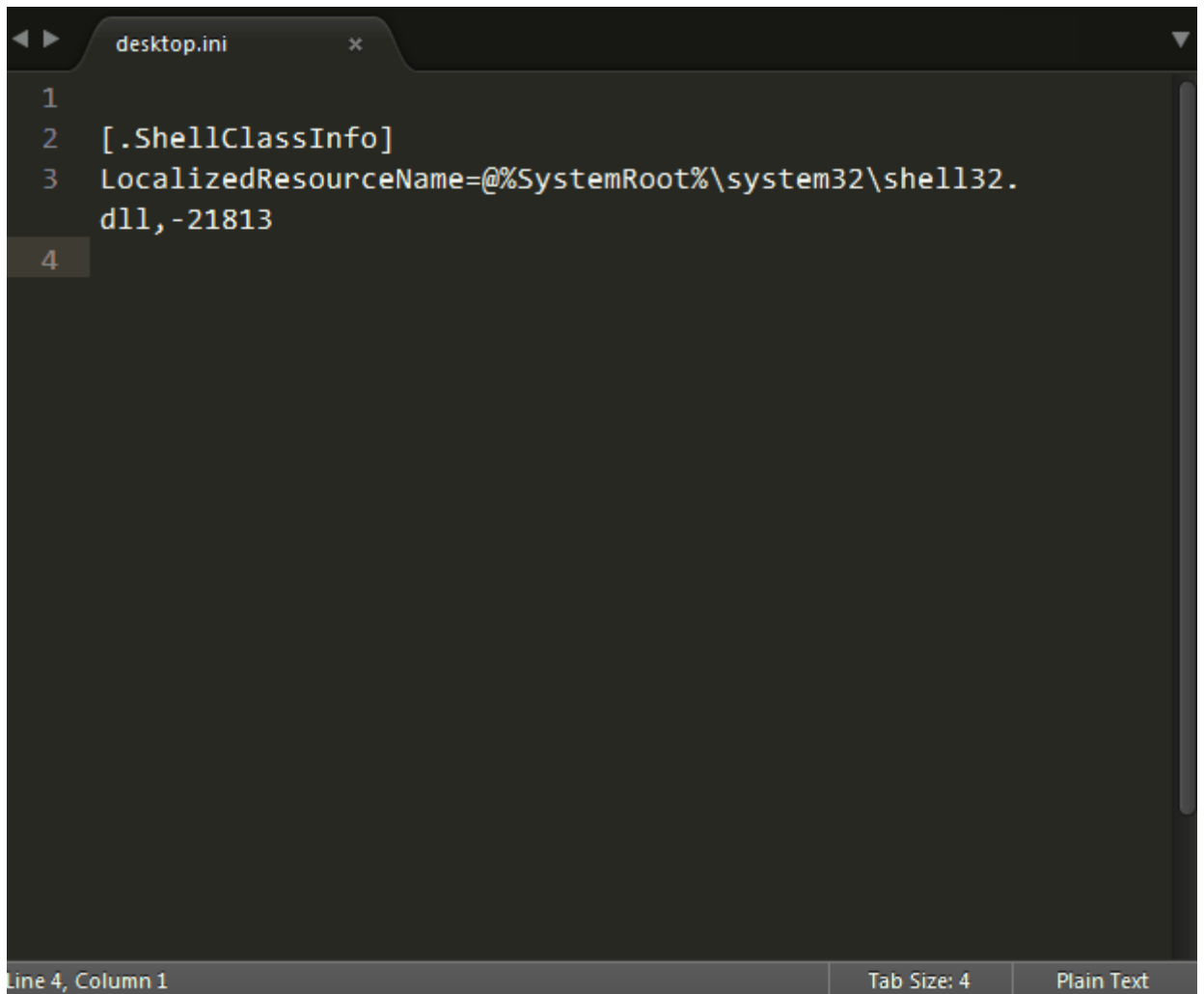
使用 **Ctrl + Shift + N** 创建一个新窗口（该快捷键再次和搜狗输入法快捷键冲突，个人建议禁用所有搜狗输入法快捷键）。

当窗口内没有标签时，使用 **Ctrl + W** 关闭该窗口。

1.7.2 标签 (Tab)

使用 **Ctrl + N** 在当前窗口创建一个新标签，**Ctrl + W** 关闭当前标签，**Ctrl + Shift + T** 恢复刚刚关闭的标签。

编辑代码时我们经常会开多个窗口，所以分屏很重要。**Alt + Shift + 2** 进行左右分屏，**Alt + Shift + 8** 进行上下分屏，**Alt + Shift + 5** 进行上下左右分屏（即分为四屏）。



```
desktop.ini
1
2 [.ShellClassInfo]
3 LocalizedResourceName=@%SystemRoot%\system32\shell32.
4 dll,-21813
```

Line 4, Column 1 Tab Size: 4 Plain Text

分屏之后，使用 **Ctrl + 数字键** 跳转到指定屏，使用 **Ctrl + Shift + 数字键** 将当前屏移动到指定屏。例如，**Ctrl + 1** 会跳转到 1 屏，而 **Ctrl + Shift + 2** 会将当前屏移动到 2 屏。

1.7.3 全屏（Full Screen）

Sublime Text 有两种全屏模式：普通全屏和无干扰全屏。

个人强烈建议在开启全屏前关闭菜单栏（Toggle Menu），否则全屏效果会大打折扣。

F11 切换普通全屏：

```
untitled  Sublime Text 3 全攻略3.md  sample.cpp  sample.py
293
294 当窗口内没有标签时，使用'Ctrl + W'关闭该窗口。
295
296 ### 标签 ( Tab )
297
298 使用'Ctrl + N'在当前窗口创建一个新标签，'Ctrl + W'关闭当前标签，'Ctrl + Shift + T'恢复刚刚关闭的标签。
299
300 编辑代码时我们经常会开多个窗口，所以分屏很重要。'Alt + Shift + 2'进行左右分屏，'Alt + Shift + 8'进行上下分屏，'Alt + Shift + 5'
    进行上下左右分屏（即分为四屏）。
301
302 ![各种分屏]()
303
304 分屏之后，使用'Ctrl + 数字键'跳转到指定屏，使用'Ctrl + Shift + 数字键'将当前屏移动到指定屏。例如，'Ctrl + 1'会跳转到1屏，而'Ctrl + Shift
    + 2'会将当前屏移动到2屏。
305
306 ### 全屏 ( Full Screen )
307
308 Sublime Text有两种全屏模式：普通全屏和无干扰全屏。
309
310 个人强烈建议在开启全屏前关闭菜单栏（Toggle Menu），否则全屏效果会大打折扣。
311
312 'F11'开启普通全屏：
313
314 ![普通全屏]()
315
316 'Shift + F11'开启无干扰全屏：
317
318 ![无干扰全屏]()
319
320 ## 风格 ( Styles )
321
322 风格对于任何软件都很重要，对编辑器也是如此，尤其是GUI环境下的编辑器。作为一个程序员，我希望我的编辑器足够简洁且足够个性。
323
324 ![Notepad++ vs. Sublime Text]()
325
326 所以在用过Sublime Text之后，我立刻就卸掉了Notepad++
```

Shift + F11 切换无干扰全屏：

```
当窗口内没有标签时，使用'Ctrl + W'关闭该窗口。

### 标签 ( Tab )

使用'Ctrl + N'在当前窗口创建一个新标签，'Ctrl + W'关闭当前标签，'Ctrl + Shift + T'
恢复刚刚关闭的标签。

编辑代码时我们经常会开多个窗口，所以分屏很重要。'Alt + Shift + 2'进行左右分屏，'
Alt + Shift + 8'进行上下分屏，'Alt + Shift + 5'进行上下左右分屏（即分为四屏）。

![各种分屏]()

分屏之后，使用'Ctrl + 数字键'跳转到指定屏，使用'Ctrl + Shift + 数字键'
将当前屏移动到指定屏。例如，'Ctrl + 1'会跳转到1屏，而'Ctrl + Shift + 2'
会将当前屏移动到2屏。

### 全屏 ( Full Screen )

Sublime Text有两种全屏模式：普通全屏和无干扰全屏。

个人强烈建议在开启全屏前关闭菜单栏（Toggle Menu），否则全屏效果会大打折扣。

'F11'开启普通全屏：

![普通全屏]()

'Shift + F11'开启无干扰全屏：

![无干扰全屏]()

## 风格 ( Styles )

风格对于任何软件都很重要，对编辑器也是如此，尤其是GUI环境下的编辑器。作为一个程
序员，我希望我的编辑器足够简洁且足够个性。

![Notepad++ vs. Sublime Text]()

所以在用过Sublime Text之后，我立刻就卸掉了Notepad++
```

1.8 风格（Styles）

风格对于任何软件都很重要，对编辑器也是如此，尤其是 GUI 环境下的编辑器。作为一个程序员，我希望我的编辑器足够简洁且足够个性。

Notepad++默认界面

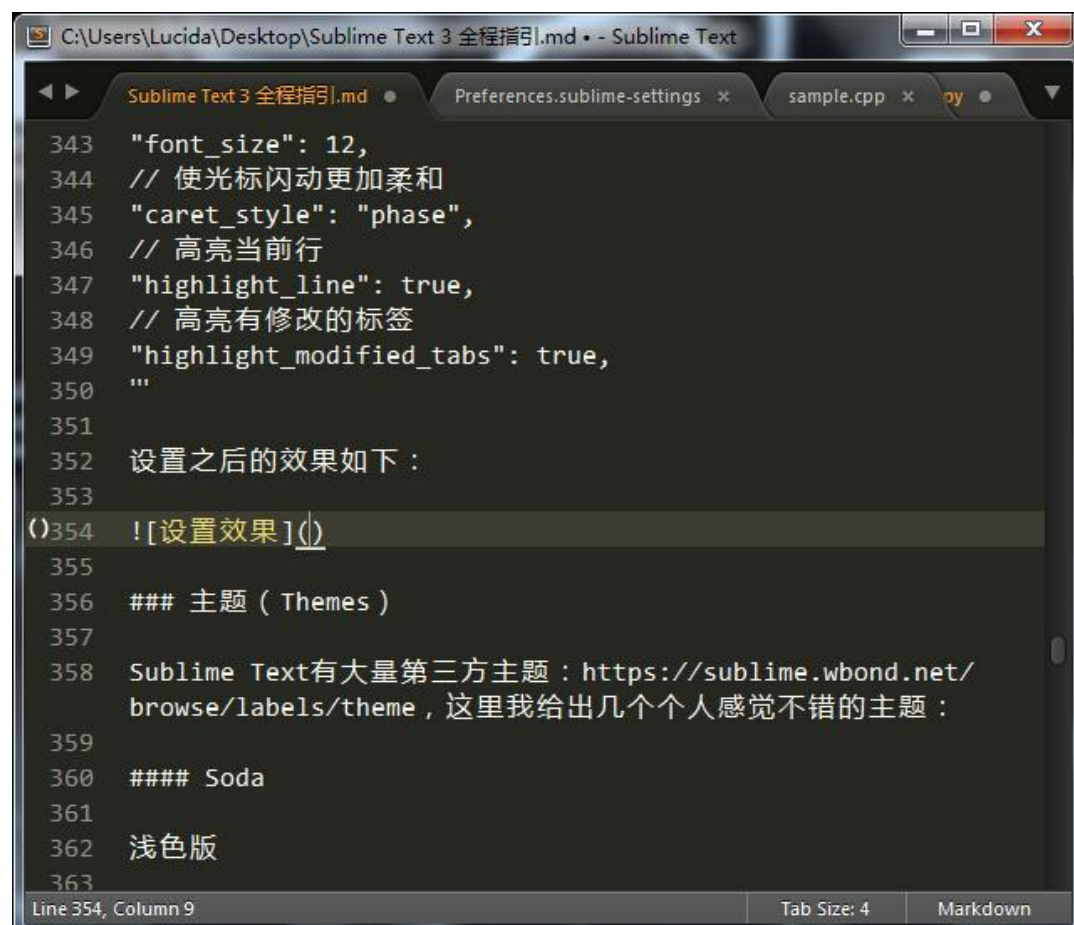
Sublime Text 自带的风格是我喜欢的深色风格（也可以调成浅色），默认主题是 **Monokai Bright**，这两者的搭配已经很不错了，不过我们还可以做得更好：接下来我将会展示如何通过设置偏好项和添加自定义风格/主题使得 Sublime Text 更加 Stylish。

1.8.1 一些设置 (Miscellaneous Settings)

下面是我个人使用的设置项。

```
// 设置 Sans-serif（无衬线）等宽字体，以便阅读
"font_face": "YaHei Consolas Hybrid",
"font_size": 12,
// 使光标闪动更加柔和
"caret_style": "phase",
// 高亮当前行
"highlight_line": true,
// 高亮有修改的标签
"highlight_modified_tabs": true,
```

设置之后的效果如下：

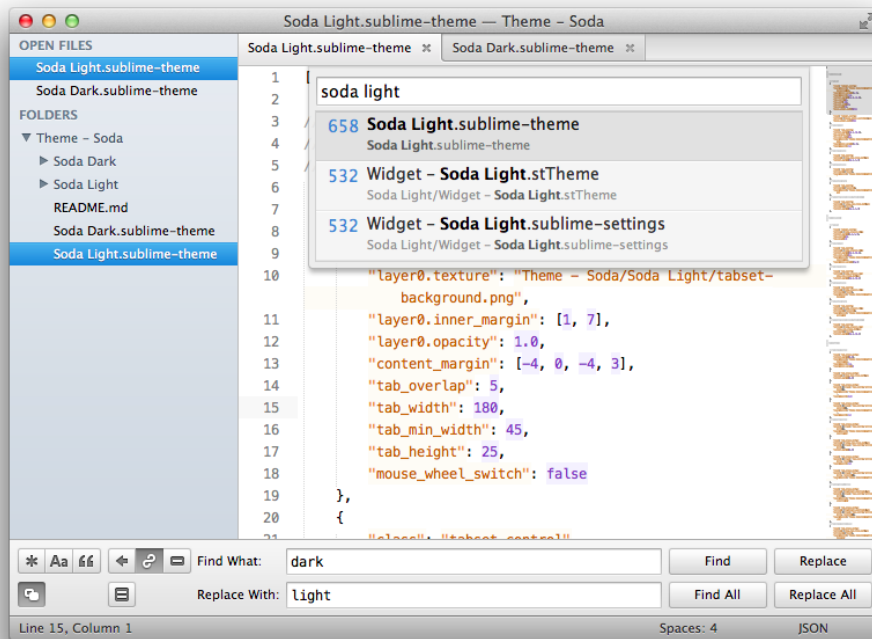


1.8.2 主题 (Themes)

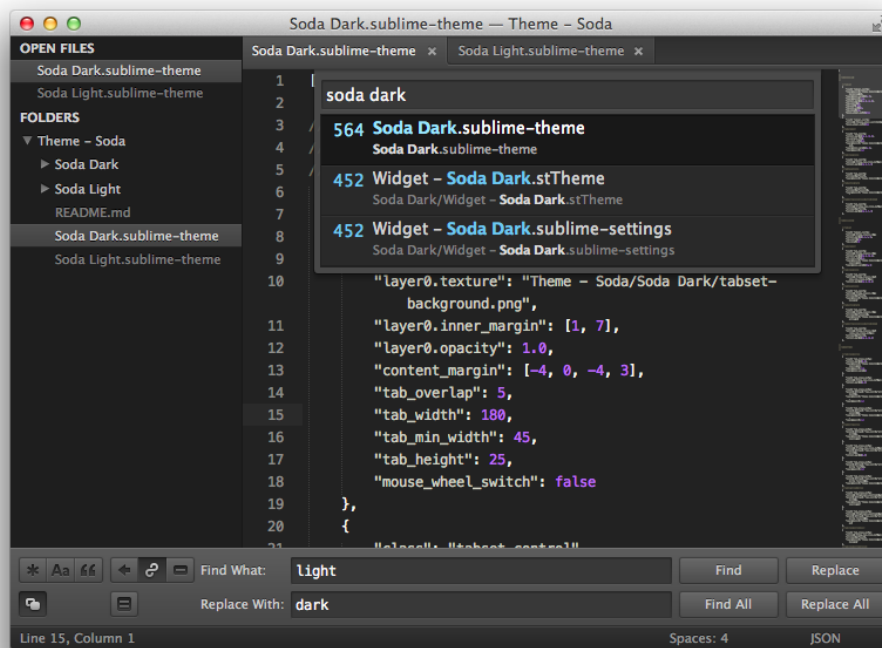
Sublime Text 有大量第三方主题：

[<https://sublime.wbond.net/browse/labels/theme>]，这里我给出几个个人感觉不错的主题：

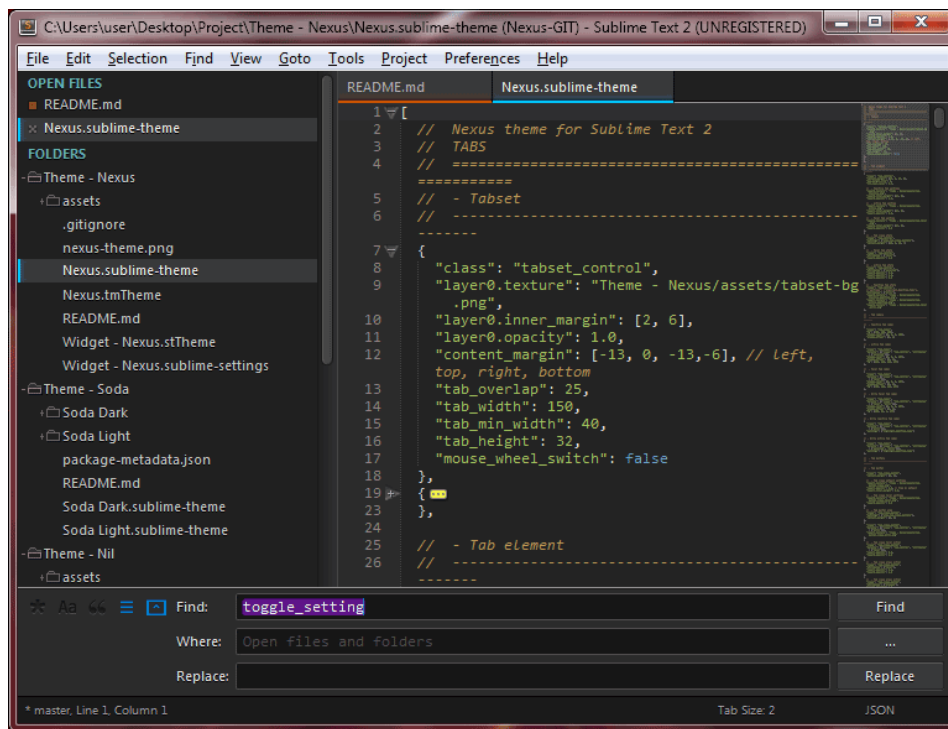
1. Soda Light



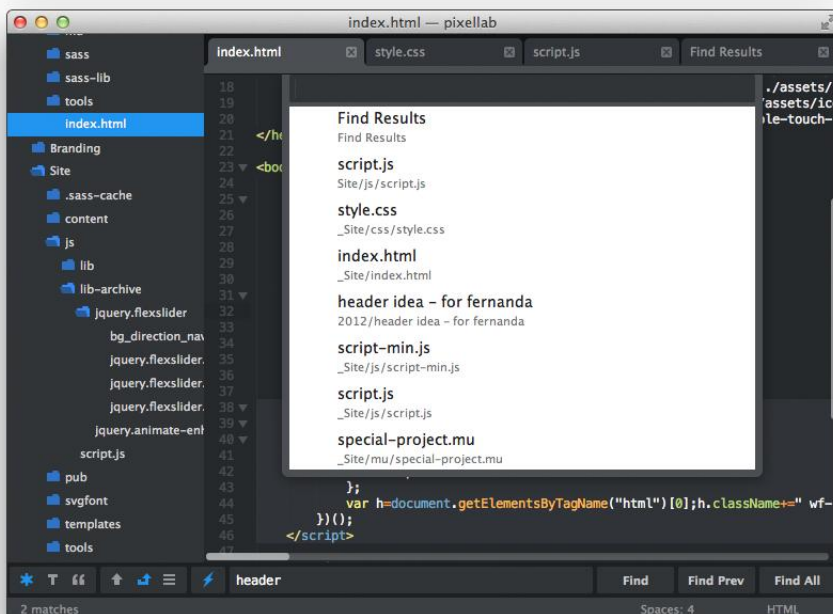
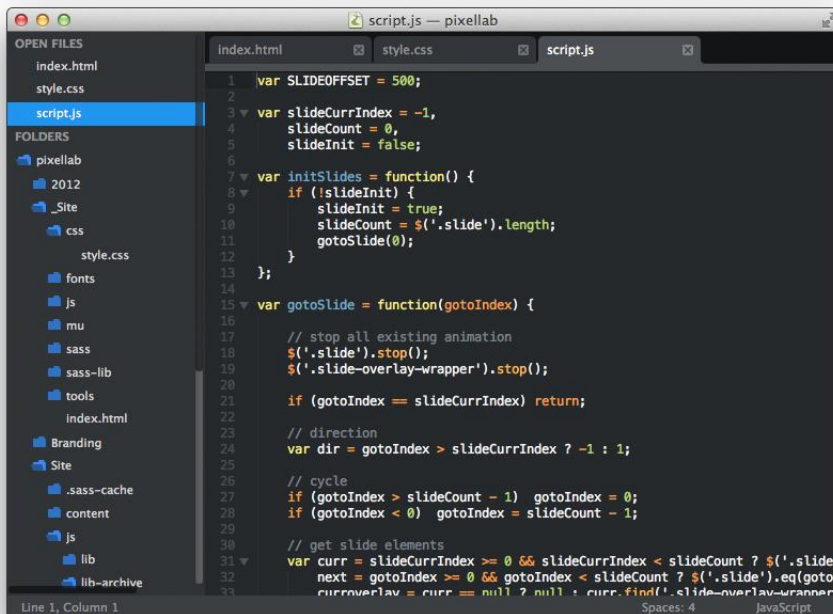
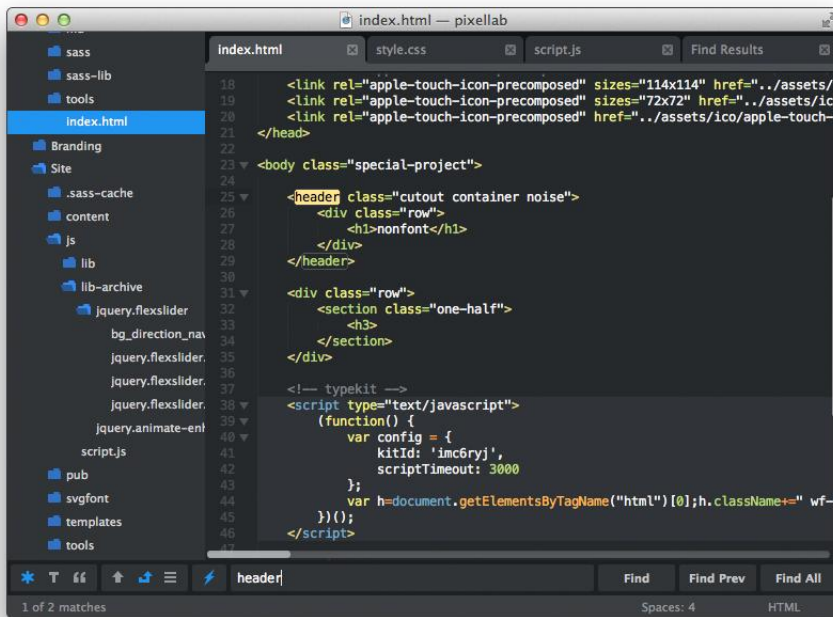
2. Soda Dark



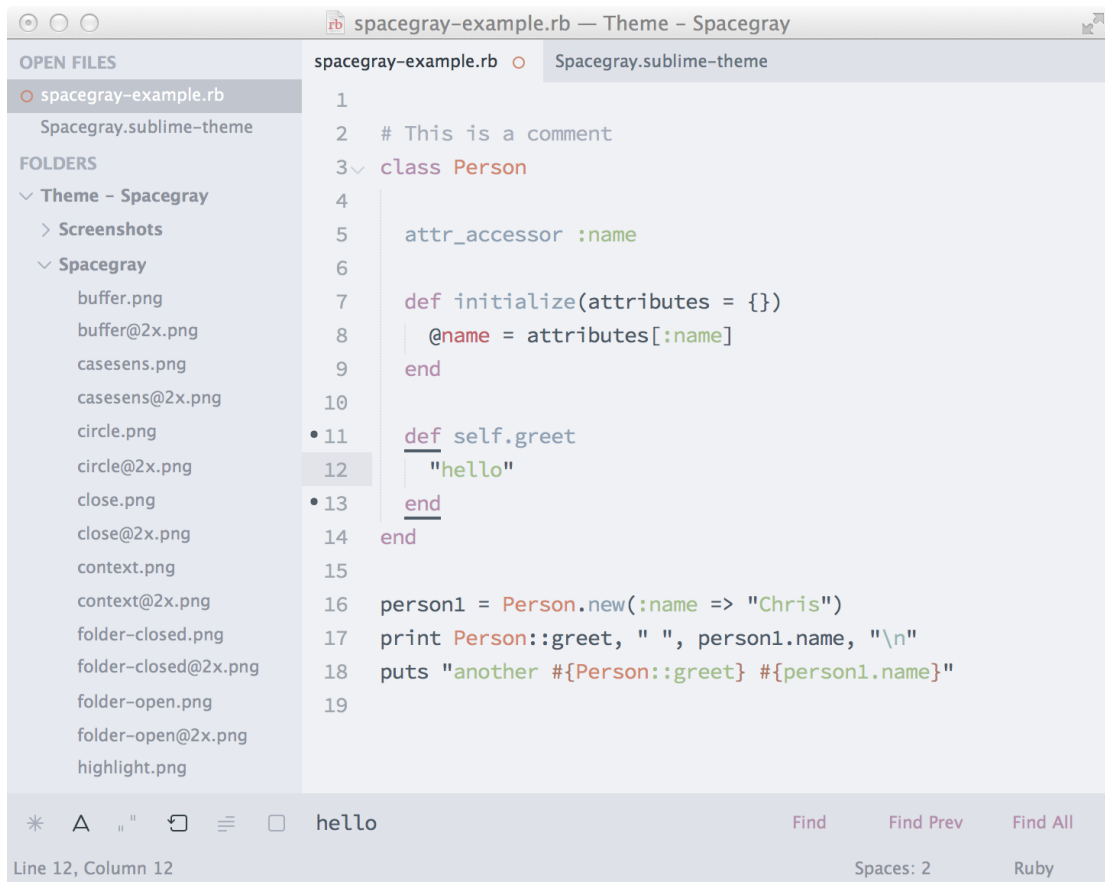
3. Nexus



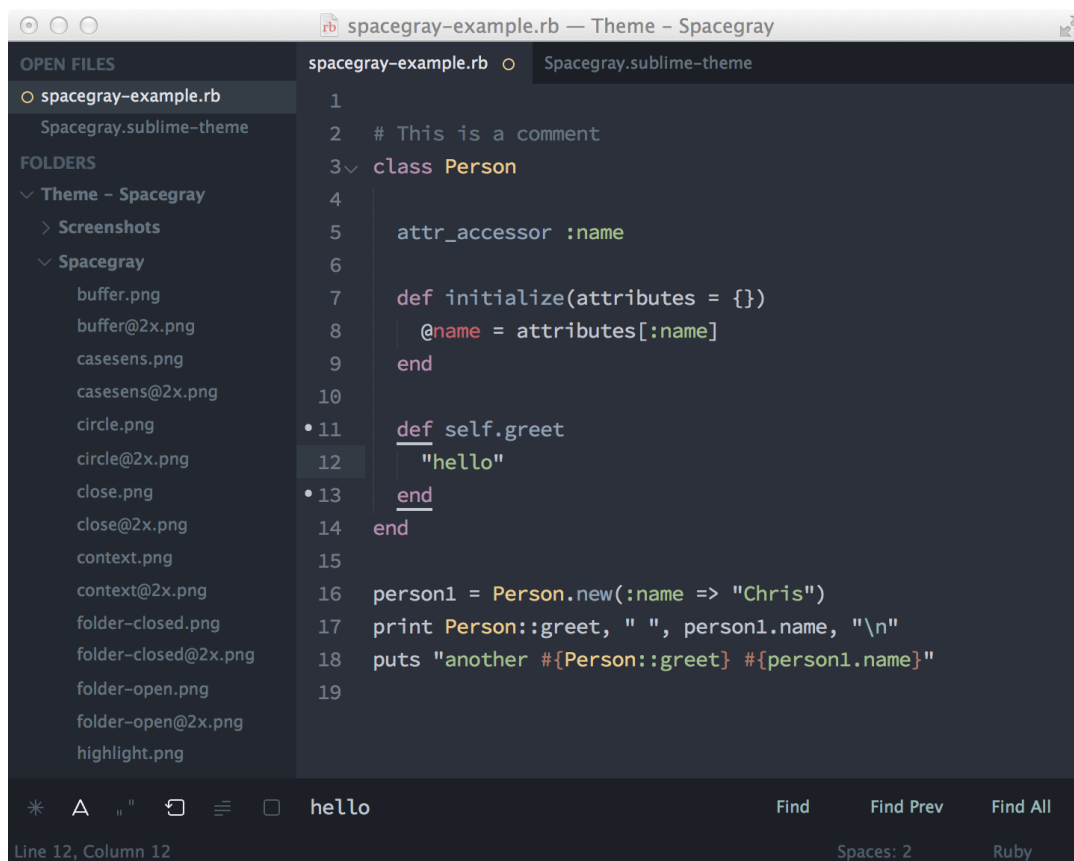
4. Flatland



5. Spacegray Light



6. Spacegray Dark



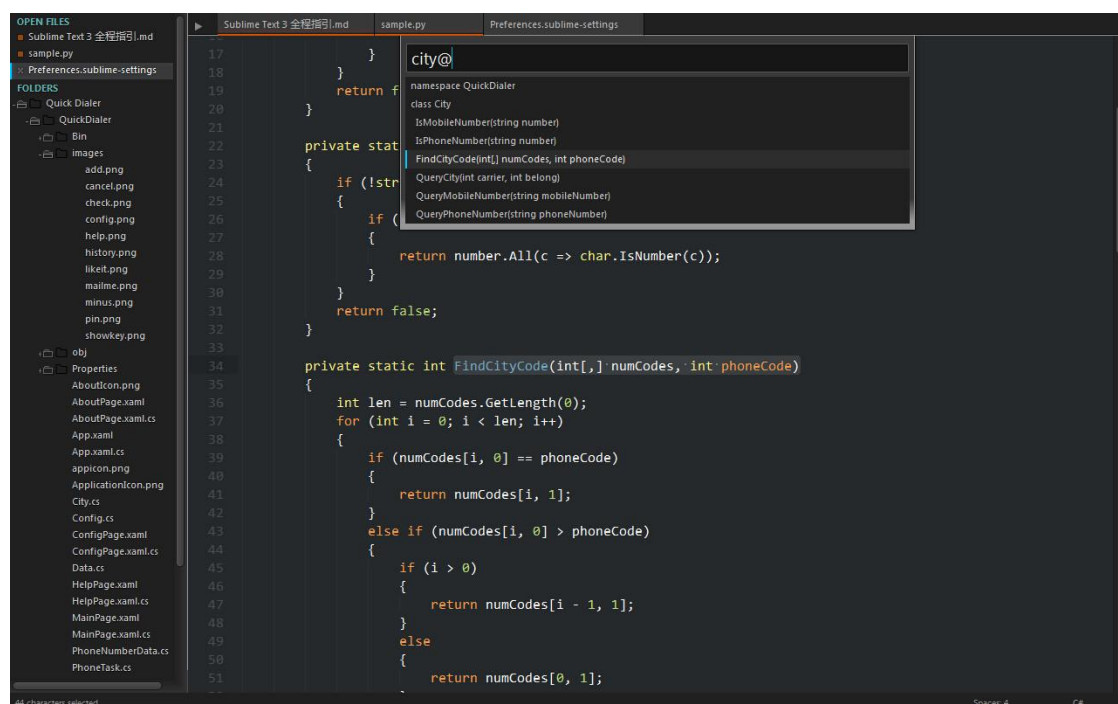
1.8.3 配色（Color）

[colorsublime](#) 包含了大量 Sublime Text 配色方案，并支持在线预览，配色方案的安装教程在[这里](#)，恕不赘述。

我个人使用的是 [Nexus](#) 主题和 [Flatland Dark](#) 配色，配置如下：

```
"theme": "Nexus.sublime-theme",
"color_scheme": "Packages/Theme - Flatland/Flatland Dark.tmTheme",
```

效果如下：



1.9 编码（Coding）

优秀的编辑器使编码变的更加容易，所以 Sublime Text 提供了一系列功能以提高开发效率。

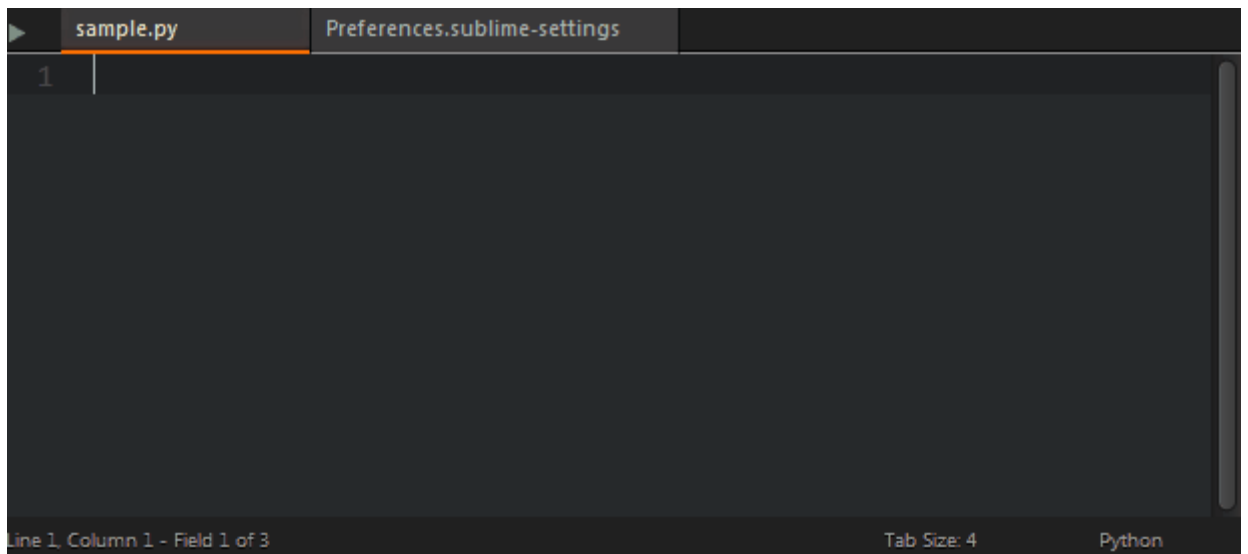
1.9.1 良好实践（Good Practices）

良好的代码应该是规范的，所以 Google 为每一门主流语言都设置了其代码规范（Code Style Guideline）。我自己通过下面的设置使以规范化自己的代码。

```
// 设置 tab 的大小为 2
"tab_size": 2,
// 使用空格代替 tab
"translate_tabs_to_spaces": true,
// 添加行宽标尺
"rulers": [80, 100],
// 显示空白字符
"draw_white_space": "all",
// 保存时自动去除行末空白
"trim_trailing_white_space_on_save": true,
// 保存时自动增加文件末尾换行
"ensure_newline_at_eof_on_save": true,
```

1.9.2 代码段（Code Snippets）

Sublime Text 支持代码段（Code Snippet），输入代码段名称后 **Tab** 即可生成代码段。



你可以通过 Package Control 安装第三方代码段，也可以自己创建代码段，参考[这里](#)。

1.9.3 格式化（Formatting）

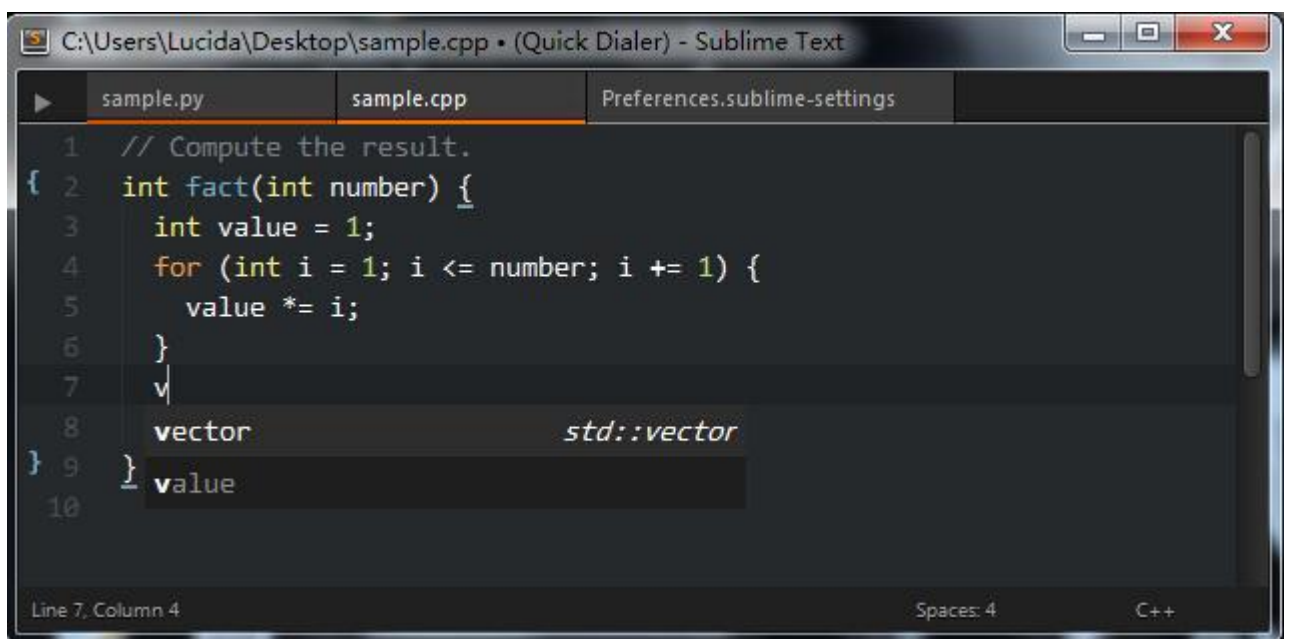
Sublime Text 基本的手动格式化操作包括：**Ctrl + [**向左缩进，**Ctrl +]**向右缩进，此外**Ctrl + Shift + V**可以以当前缩进粘贴代码（非常实用）。

除了手动格式化，我们也可以通过安装插件实现自动缩进和智能对齐：

- **HTMLBeautify**: 格式化 HTML。
- **AutoPEP8**: 格式化 Python 代码。
- **Alignment**: 进行智能对齐。

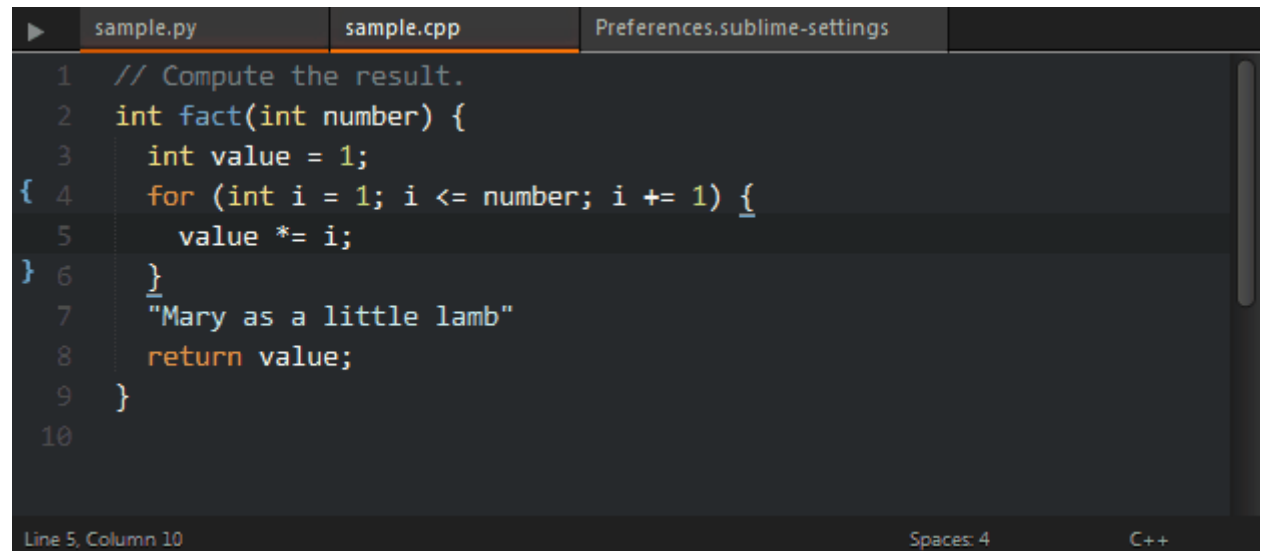
1.9.4 自动完成（Auto Completion）

Sublime Text 支持一定的自动完成，按 **Tab** 自动补全。



1.9.5 括号 (Brackets)

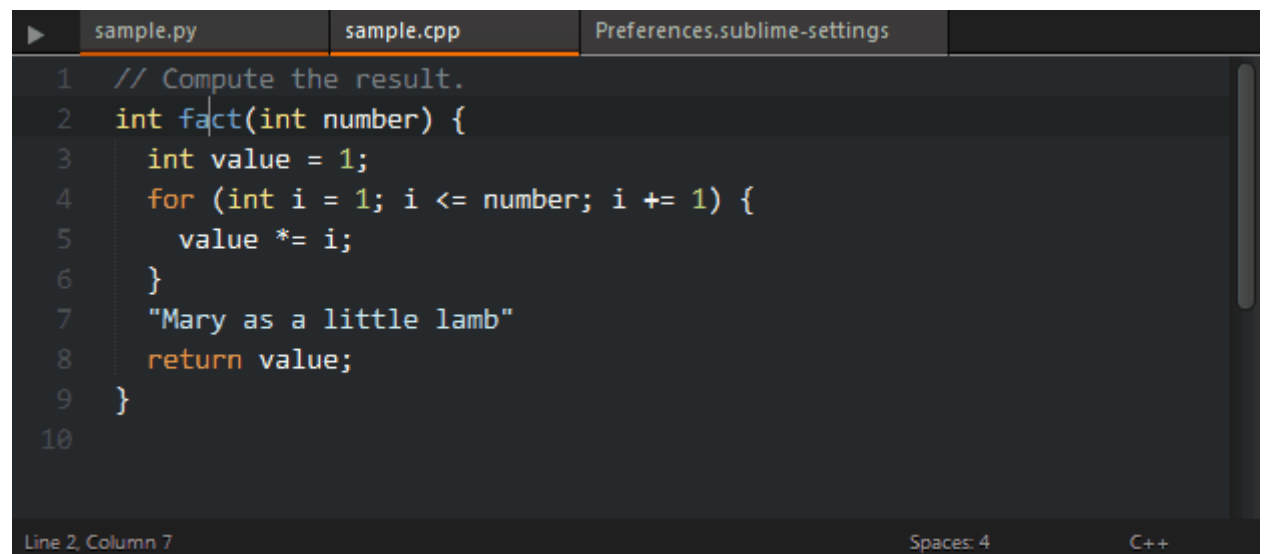
编写代码时会碰到大量的括号，利用 **Ctrl + M** 可以快速的在起始括号和结尾括号间切换，**Ctrl + Shift + M** 则可以快速选择括号间的内容，对于缩进型语言（例如 Python）则可以使用 **Ctrl + Shift + J**。



```
1 // Compute the result.
2 int fact(int number) {
3     int value = 1;
4     for (int i = 1; i <= number; i += 1) {
5         value *= i;
6     }
7     "Mary as a little lamb"
8     return value;
9 }
10
```

Line 5, Column 10 Spaces: 4 C++

此外，我使用 BracketHighlighter 插件以高亮显示配对括号以及当前光标所在区域，效果如下：

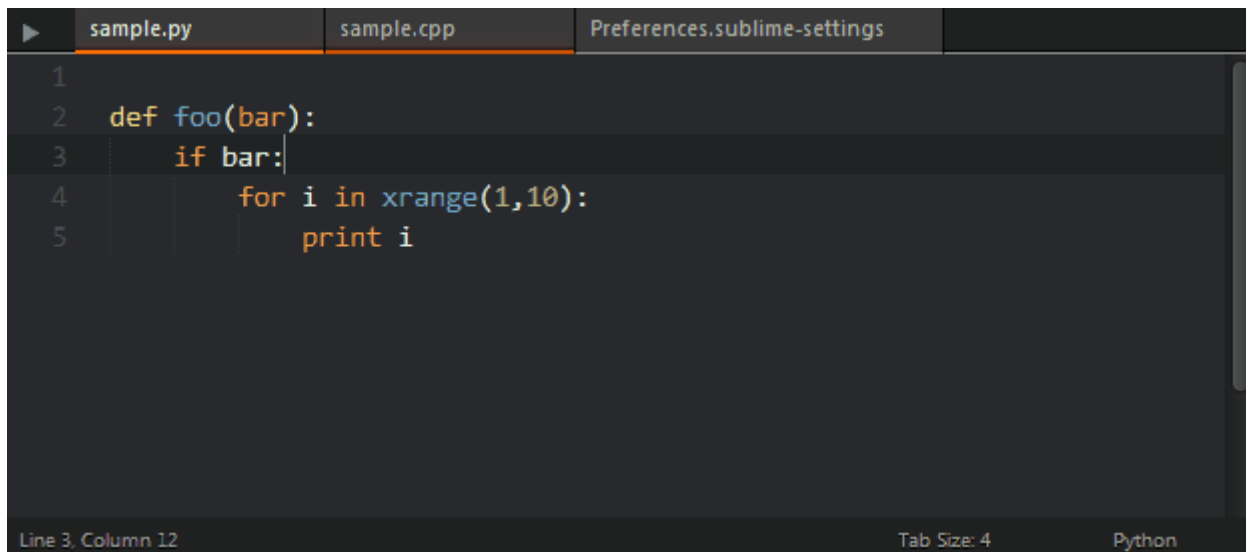


```
1 // Compute the result.
2 int fact(int number) {
3     int value = 1;
4     for (int i = 1; i <= number; i += 1) {
5         value *= i;
6     }
7     "Mary as a little lamb"
8     return value;
9 }
10
```

Line 2, Column 7 Spaces: 4 C++

1.9.6 命令行 (Command Line)

尽管提供了 Python 控制台，但 Sublime Text 的控制台仅支持单行输入，十分不方便，所以我使用 SublimeREPL 以进行一些编码实验 (Experiments)。



```
1
2 def foo(bar):
3     if bar:
4         for i in xrange(1,10):
5             print i
```

Line 3, Column 12 Tab Size: 4 Python

1.10 其它 (Miscellaneous)

尽管我试图在本文包含尽可能多的 Sublime Text 实用技能，但受限于篇幅和我的个人经验，本文仍不免有所遗漏，欢迎在评论里指出本文的错误及遗漏。

下面是一些可能有用但我很少用到的功能：

- 宏 (Macro)：Sublime Text 支持[录制宏](#)，但我在实际工作中并未发现宏有多大用处。
- 其它平台 (Other Platforms)：本文只介绍了 Windows 平台上 Sublime Text 的使用，不过 Linux 和 OS X 上 Sublime Text 的使用方式和 Windows 差别不大，只是在快捷键上有所差异，请参考 [Windows/Linux 快捷键](#) 和 [OS X 快捷键](#)。
- 项目 (Projects)：Sublime Text 支持简单的[项目管理](#)，但我一般只用到文件夹。
- Vim 模式 (Vintage)：Sublime Text 自带 [Vim 模式](#)。
- 构建 (Build)：通过配置，Sublime Text 可以进行[源码构建](#)。
- 调试 (Debug)：通过安装[插件](#)，Sublime Text 可以对代码进行调试。

1.11 快捷键列表 (Shortcuts Cheatsheet)

我把本文出现的 Sublime Text 按其类型整理在这里，以便查阅。

1.11.1 通用 (General)

- **↑ ↓ ← →**：上下左右移动光标，注意不是不是 **KJHL**！
- **Alt**：调出菜单
- **Ctrl + Shift + P**：调出命令板 (Command Palette)
- **Ctrl + `**：调出控制台

1.11.2 编辑 (Editing)

- **Ctrl + Enter**：在当前行下面新增一行然后跳至该行

- **Ctrl + Shift + Enter**: 在当前行上面增加一行并跳至该行
- **Ctrl + ←/→**: 进行逐词移动
- **Ctrl + Shift + ←/→** 进行逐词选择
- **Ctrl + ↑/↓** 移动当前显示区域
- **Ctrl + Shift + ↑/↓** 移动当前行

1.11.3 选择 (Selecting)

- **Ctrl + D**: 选择当前光标所在的词并高亮该词所有出现的位置，再次 **Ctrl + D** 选择该词出现的下一个位置，在多重选词的过程中，使用 **Ctrl + K** 进行跳过，使用 **Ctrl + U** 进行回退，使用 **Esc** 退出多重编辑
- **Ctrl + Shift + L**: 将当前选中区域打散
- **Ctrl + J**: 把当前选中区域合并为一行
- **Ctrl + M**: 在起始括号和结尾括号间切换
- **Ctrl + Shift + M**: 快速选择括号间的内容
- **Ctrl + Shift + J**: 快速选择同缩进的内容
- **Ctrl + Shift + Space**: 快速选择当前作用域 (Scope) 的内容

1.11.4 查找&替换 (Finding&Replacing)

- **F3**: 跳至当前关键字下一个位置
- **Shift + F3**: 跳到当前关键字上一个位置
- **Alt + F3**: 选中当前关键字出现的所有位置
- **Ctrl + F/H**: 进行标准查找/替换，之后：
 - **Alt + C**: 切换大小写敏感 (Case-sensitive) 模式
 - **Alt + W**: 切换整字匹配 (Whole matching) 模式
 - **Alt + R**: 切换正则匹配 (Regex matching) 模式
 - **Ctrl + Shift + H**: 替换当前关键字
 - **Ctrl + Alt + Enter**: 替换所有关键字匹配
- **Ctrl + Shift + F**: 多文件搜索&替换

1.11.5 跳转 (Jumping)

- **Ctrl + P**: 跳转到指定文件，输入文件名后可以：
 - **@ 符号跳转**: 输入 **@symbol** 跳转到 **symbol** 符号所在的位置
 - **# 关键字跳转**: 输入 **#keyword** 跳转到 **keyword** 所在的位置
 - **: 行号跳转**: 输入 **:12** 跳转到文件的第 12 行。
- **Ctrl + R**: 跳转到指定符号
- **Ctrl + G**: 跳转到指定行号

1.11.6 窗口 (Window)

- **Ctrl + Shift + N**: 创建一个新窗口

- **Ctrl + N**: 在当前窗口创建一个新标签
- **Ctrl + W**: 关闭当前标签，当窗口内没有标签时会关闭该窗口
- **Ctrl + Shift + T**: 恢复刚刚关闭的标签

1.11.7 屏幕 (Screen)

- **F11**: 切换普通全屏
- **Shift + F11**: 切换无干扰全屏
- **Alt + Shift + 2**: 进行左右分屏
- **Alt + Shift + 8**: 进行上下分屏
- **Alt + Shift + 5**: 进行上下左右分屏
- 分屏之后，使用 **Ctrl + 数字键** 跳转到指定屏，使用 **Ctrl + Shift + 数字键** 将当前屏移动到指定屏

1.12 延伸阅读 (Further Reading)

1.12.1 书籍 (Books)

- Mastering Sublime Text: 我读过的唯一一本关于 Sublime Text 的书籍，书中介绍的插件很实用，但对编辑技巧介绍不全。
- Instant Sublime Text Starter: 另外一本关于 Sublime Text 的书，我没有读过。

1.12.2 链接 (Links)

- 官方文档: <http://www.sublimetext.com/docs/3/>
- 官方论坛: <http://www.sublimetext.com/forum/>
- Stack Overflow 的 Sublime Text 频道:
- <http://stackoverflow.com/questions/tagged/sublimetext>
- <http://stackoverflow.com/questions/tagged/sublimetext2>
- <http://stackoverflow.com/questions/tagged/sublimetext3>
- 非官方文档: <http://sublime-text-unofficial-documentation.readthedocs.org/> 甚至比官方文档还要全面!
- Package Control: <https://sublime.wbond.net/> 大量的 Sublime Text 插件和主题。

1.12.3 视频 (Videos)

Getting Started with SublimeText :
<https://www.youtube.com/watch?v=04gKiTiRlq8>

Sublime Text Pefect Workflow :
<https://www.youtube.com/watch?v=bpEp0ePIOEM&list=PLuwqxbyf3olpLsnFvo06qbrkcEB5o7K0g>