

Pycharm使用秘笈v0.3

笔记本： Python Dedication

创建时间： 2016/7/31 22:55

更新时间： 2017/5/1 20:36

作者： 466485959@qq.com

PyCharm使用秘籍

1. PyCharm的基本使用

1. 在PyCharm下为你的Python项目配置Python解释器
 1. Project:当前项目名>Project Interpreter>add Local
2. 在PyCharm下创建Python文件、Python模块
 1. File>New>Python File
 2. File>New>Python Package
3. 使用PyCharm安装Python第三方模块
 1. Project:当前项目名>Project Interpreter>点击右侧绿色小加号
4. PyCharm基本设置，例如不使用tab、tab=4空格、字体、字体颜色、主题、脚本头设置、显示行号等。如何导出和导入自定义设置。
 1. 设置菜单字体大小:Appearance&Behavior>Appearance
 2. 设置Console&Terminal字体大小: Editor>Colors&Fonts>Console Font
 3. 设置文件编码：Editor>File Encodings IDE Encoding; Project Encoding;Properties Files都设置为UTF-8
 4. 修改背景颜色：Settings>Editor>Color&Fonts>General>右边Text下面选中Default text>修改Background颜色即可
 5. 不使用tab、tab=4空格：Editor>Code Style>Python
 6. 字体、字体颜色：Edit>Colors & Fonts>Python
 7. 关闭自动检查更新：Appearance & Behavior>System Settings>Updates
 8. 脚本头设置：Edit>File and Code Templates>Python Script
 1. 控制光标位置：#[[\$END\$]]#（勾选Enable Live Templates）
 2. 其他类似
 9. 显示行号：Edit>General>Appearance>Show line numbers 注：2016.2默认显示行号
 10. 右侧竖线是PEP8的代码规范，提示一行不要超过120个字符
 11. 导出、导入你自定义的Pycharm配置：File>Export Settings、Import Settings
5. 常用快捷键，例如复制当前行、删除当前行、批量注释、缩进、查找和替换。
 1. 常用快捷键的查询和配置：Keymap
 1. Ctrl + D：复制当前行
 2. Ctrl + E：删除当前行
 3. Shift + Enter：快速换行
 4. Ctrl + /：快速注释（选中多行后可以批量注释）
 5. Tab：缩进当前行（选中多行后可以批量缩进）
 6. Shift + Tab：取消缩进（选中多行后可以批量取消缩进）
 7. Ctrl + F：查找
 8. Ctrl + H：替换
 9. Ctrl + 减号：折叠当前代码块;Ctrl + 加号：展开当前代码块
 10. Ctrl + Shift + 减号：折叠当前文件；Ctrl + Shift + 加号：展开当前文件
 2. 设置快捷键
 1. 设置修改字体大小的快捷键
6. PyCharm安装插件

1. Plugins>Browse repositories (下方三个按钮中间那个)>搜索 'markdown support' >install。右上角View有三个选项可选。
2. [.ignore](#)
3. [BashSupport](#)
4. [IdeaVim](#)
5. [CodeGlance](#)
7. Git配置
 1. 需要本地安装好Git
 2. Version Control>Git
 3. 配置了Git等版本控制系统之后，可以实时查看diff、本地回滚修改等
 4. [配置github : Settings>Version Control>GitHub>右侧填写Host、Login及Password即可。](#)
8. 常用操作指南。例如复制文件路径、在文件管理器中打开、快速定位、查看模块结构视图、tab批量换space、TODO的使用、Debug的使用。
 1. 复制文件路径：左侧文件列表右键选中的文件>Copy Path
 2. 在文件管理器中打开：右键选中的文件>往下找到Show In Explorer
 3. 快速定位：Ctrl + 某些内建模块之后，点击在源文件中展开
 4. 查看结构：IDE左侧边栏Structure 查看当前项目的结构
 5. tab批量换space：Edit>Convert Indents
 6. TODO的使用：# TODO 要记录的事情，commit之前会有提示
 7. [Debug设置断点，直接点击行号与代码之间的空白处即可设置断点，debug一般只需在关键点设置一个，然后debug调试时步进执行。没必要点很多个断点](#)
 8. Tab页上右键>Move Right (Down)，把当前Tab页移到窗口右边 (下边)，方便对比
 9. 文件中右键>Local History能够查看文件修改前后的对比
 10. IDE右下角能看到一些有用的信息，光标当前在第几行的第几个字符、当前回车换行、当前编码类型、当前Git分支
 11. [右边栏Database \(点开 \)>左上角绿色 "+" 号>下拉Data Source选择你要连接的数据库类型>点击之后页面最下方会有提示安装驱动 \(Download missing driver files \)](#)
 12. [执行某个文件中的某一行 \(某些行 \)：选中要执行的代码部分>右键Execute Selection in Console](#)
9. 如何去掉烦人的波浪线？PEP8又是什么？
 1. settings>Editor>Inspections>Python
 2. 单独一行的注释：#+1空格+注释内容
 3. 代码后跟着的注释：2空格+#+1空格+注释内容
 4. [函数前面空一行，类前面空两行](#)
 5. [某些单词一直有下划线提示，可以右键>Spelling>Typo:Save 'xxx' to dictionary](#)
10. SSH Terminal：Default encoding:UTF-8
 1. Settings>Tools>SSH Terminal>最后一行Default encoding:选择UTF-8
11. [Live Templates \(善用live templates提高开发效率 \)](#)
 1. Settings>Editor>Live Templates
12. 远程调试！边改边同步到远程服务器，本地直接执行远程服务器上的代码！
 1. [Build,Execution,Deployment>Deployment>点击绿色 "+" 添加一个Deployment配置。配置好SFTP之后可以右键上传更新后的代码文件。](#)
 2. [Project Interpreter>Add Remote>选择Deployment configuration>下拉框选择上面的配置>下面选择python解释器路径](#)
 3. [关掉对话框，配置Path mappings。](#)
 4. [参考链接：官网文档：https://www.jetbrains.com/help/pycharm/2017.1/configuring-remote-](#)

[interpreters-via-deployment-configuration.html?search=remote](https://www.jetbrains.com/idea/faq/interpreters-via-deployment-configuration.html?search=remote)

13. 善用搜索！善用搜索！善用搜索！

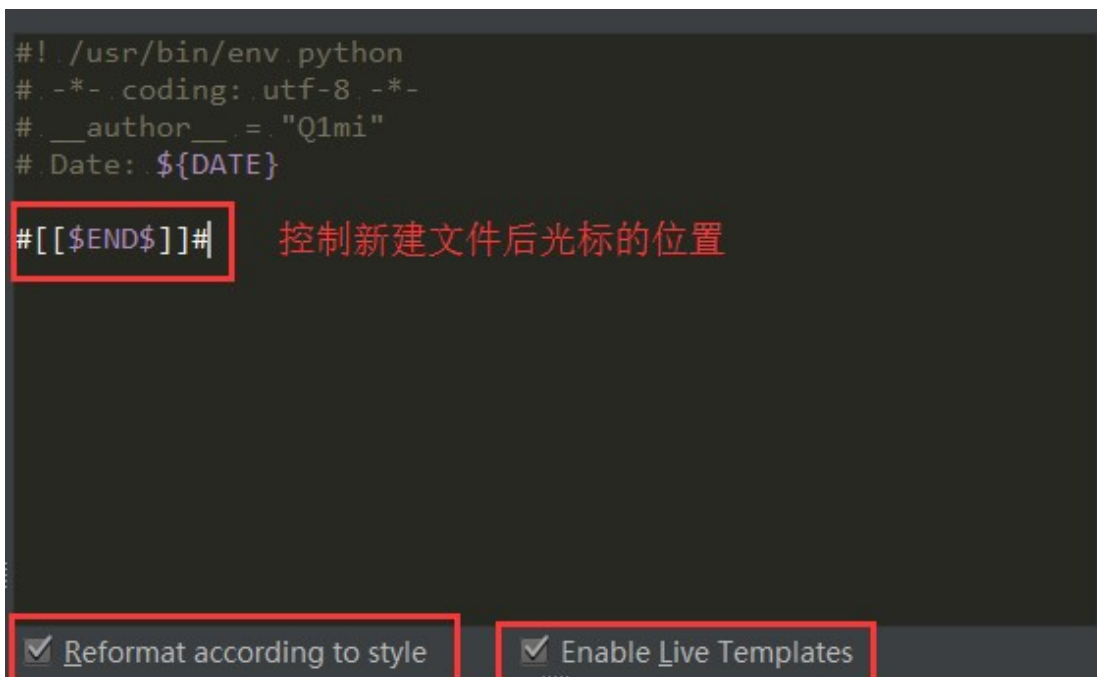
1. 遇到不懂问题，先尝试自己解决。这样才记得更牢！

file and code template 可用的变量<https://www.jetbrains.com/help/pycharm/2016.3/file-and-code-templates-2.html>：

- Variables to be expanded into corresponding values in the `${<variable_name>}` format.

The available predefined file template variables are:

- `${PROJECT_NAME}` - the name of the current project.
- `${NAME}` - the name of the new file which you specify in the **New File** dialog box during the file creation.
- `${USER}` - the login name of the current user.
- `${DATE}` - the current system date.
- `${TIME}` - the current system time.
- `${YEAR}` - the current year.
- `${MONTH}` - the current month.
- `${DAY}` - the current day of the month.
- `${HOUR}` - the current hour.
- `${MINUTE}` - the current minute.
- `${PRODUCT_NAME}` - the name of the IDE in which the file will be created.
- `${MONTH_NAME_SHORT}` - the first 3 letters of the month name. Example: Jan, Feb, etc.
- `${MONTH_NAME_FULL}` - full name of a month. Example: January, February, etc.



```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
# __author__ = "Q1mi"
# Date: ${DATE}

#[[$END$]]#
```

控制新建文件后光标的位置

☒ Reformat according to style ☒ Enable Live Templates