Python 编码风格建议

痛点

- 写出狗屎代码给别人
- 接手别人的狗屎代码
- 集成代码花费时间长

目的

- 提高代码可读性
- 降低项目集成和维护难度

使用虚拟环境

- venv

  python3 -m venv py3env

  source py3env/bin/activate

- virtualenv

  virtualenv -p python3 py3env

  source py3env/bin/activate

- conda

  conda create -n py3env python=3.6

  source activate py3env

给 Python 脚本加 Shebang

添加在 Python 脚本文件的第 1 行
增加可执行权限
执行

把脚本文件所在目录加入
PATH 环境变量

Python

```
#!/usr/bin/env python3


def main():
    print('hello')


if __name__ == '__main__':
    main()
```

Shell(修改权限，执行脚本)

```
chmod +x script.py
./script.py
```

编码声明

Python3 默认使用 UTF-8 编码

使用默认编码时不需要声明

写在 Shebang 行之下

举例:

```
# -*- encoding: utf-8 -*-
# -*- encoding: latin-1 -*-
# -*- encoding: gbk -*-
```

文档字符串 (docstrings) 与注释

文档字符串使用 3 双引号 (“”“)
在 PyCharm 中配置 Google 风格

使用井号 (#) 注释
使用 3 引号临时注释大段代码

- 模块 docstring
- 类 docstring
- 方法和函数 docstring

标识符命名

项目要有统一的命名风格

- 类名使用 CamelCase 风格，例如 (Dialogmanager)
- 常量全大写，用下划线分割单词，例如 (TIMEOUT)
- 变量全小写，用下划线分割单词，例如 (model_name)
- 函数 / 方法名和变量写法一致，例如 (parse_args)

- 避免使用不易分辨的字

  如 I(大写 i) 和 l(小写 L)

  O(大写 o) 和 0(零)

字符串

- 首选单引号字符串
- 不宜超过 79 个半角字符宽度
- 使用括号拆分长字符串

- 使用括号拆分长字符串

```
long_string = ('2006 年 9 月 27 日 20 时 20 分，接马平报'
               '称其佳被盗。犯罪分子翻墙进屋，受害人'
               '上午 10 时 30 分离家，11 时 25 分回家发'
               '现家中被盗，丢失 UT 斯达康小灵通一部，'
               '2003 年购买，购价 498 元；一条黄金'
               '项链，一副黄金耳环，一枚黄金戒指，'
               '一共 10 克左右，丢失物品现金总价值'
               '2000 元。')
```

空白行

- 顶级函数和类块上下使用两个空白行
- 文档末尾使用一个空白行

使用 import 导入对象

- 分清相对导入和绝对导入 (import xxx 和 from . import yyy)
- 一个 import 语句导入一个对象

  例如 import os

  import sys
- from ... import 语句导入多个对象 (例如 from os.path import join, realpath)
- 慎用通配符导入 (from ... import *)

更多导入举例参考 http://in.deeplycurious.ai/source/python_tips/browse/master/python_code_tips.pptx

导入顺序

- 标准库
- 第三方库
- 自己开发的本地库

空格和缩进

使用 4 个空格缩进语句，缩进要对齐，不使用 tab

不使用空格的地方

- 行尾
- 左括号右侧和右括号左侧

  错误的例子 `( 'alpha', 'beta' )`
- 逗号、分号和冒号左侧

  错误的例子 `alpha, beta = 1 , 2`
- 函数定义中默认参数等号的两侧

  错误的例子 `def absolute(num = 0):`
- 函数调用时关键字参数等号的两侧

  错误的例子 `absolute(num = 0)`
- 函数调用、取索引之类紧跟左括号的情况

  错误的例子 `absolute (num=0) ; d ['key']`

使用空格的地方

- 作为分隔符的逗号右侧

  例如 `lst = [1, 2, 3]`
- 切片操作冒号两边是表达式

  例如 `arr[left+1 : right-1]`
- 赋值语句中等号两侧

  例如 `a, b = 1, 2`
- 不同优先级的算符、括号等组成的表达式中

  例如 `ans = (a+2) * (b+3)`

使用类型注解

- 搞明白函数参数和返回值
- IDE 自动补全、参数类型检查

示例

```
def sum(seq: list) -> int:
    total = 0
    for i in seq:
        total += i
    return total
```

参考：
https://www.python.org/dev/peps/pep-0484/
https://www.python.org/dev/peps/pep-3107/

PyCharm 快捷键推荐

- 自动格式化代码 (Ctrl+Alt+L)
- 自动整理导入 (Ctrl+Alt+O)
- 导入缺少的库 (Alt+Enter)

参考资料

- Python documentation

  https://docs.python.org/3/

- Google Python Style Guide

  https://google.github.io/styleguide/pyguide.html

  中文翻译 https://www.runoob.com/w3cnote/google-python-styleguide.html

- PEP 8 – Style Guide for Python Code

  https://www.python.org/dev/peps/pep-0008/

- PyCharm Help

  https://www.jetbrains.com/help/pycharm/meet-pycharm.html