

工程代码规范

Version 1.7
Kensho金融组

1. 编程环境

- python版本不低于3.6。
- java版本不低于8。
- 为培养良好的模块化设计习惯，不允许使用Jupyter notebook。

2. 项目文件组织规范

- 功能独立的模块应放置于独立文件夹中作为module。
- 使用公共定义的模块和函数，避免重复。

3. python代码规范的基本原则

- (1) 文件名的命名应该可以“顾名思义”，且符合第9条的规定。
- (2) 文件头增加编码设定：#coding: utf8
- (3) 文件头上应该有脚本作者以及工作邮箱：#author: Tian Xia (summer.xia1@pactera.com).
- (4) 每个文件应该有“伪主函数”模块，用于测试或者作为函数入口：

```
if __name__ == "__main__":  
    your-codes.
```

对于java，此条目要求为每个java文件必须有Main函数作为测试。

- (5) 采用空格代替\t。
- (6) 每个缩进占2个空格。
- (7) 每行字符不超过80个。
- (8) 连续空行不超过1个。
- (9) 类命名，每个单词的首字母大写；多个单词中间不加下划线。
- (10) 函数、变量命名：多个单词之间用下划线连接；除非是专有名词，单词一律用小写。

例如，`get_value(...); current_time = time.time()`。

对于java，此条目为：`getValue(...); currentTime = time.time()`。

- (11) 常量采用全部大小，下划线分隔。例如，`PROJECT_DIR = "..."`。
- (12) 不鼓励单词缩写，尽量使用完整单词，超过10个字母的单词除外。
- (13) 所有定义性代码（常量、函数等）写在“伪主函数”之外，其余代码写在之内。
- (14) 双目运算符、逗号，及某些情况下需加空格。例如，

```
a = b
```

```
for a in range(1, 100)
```

```
c = data[10], 不加空格。
```

```
def merge_table(id_number, data)
```

- (15) 函数的默认参数的赋值“=”不需要加空格；以及调用函数时显示指定参数时“=”不需要加空格。例如，

```
parser.add_option('--input', default="hello.txt")
```

```
def func(a, b=None)
```

- (16) 未被调用的代码应该删除。
- (17) 被注释为#code review的部分要及时修改。
- (18) 单个源文件不超过500行。
- (19) 代码中不能使用文件的绝对路径作为内部参数。
- (20) 不能使用chdir函数。
4. git常用命令，请放在 ~/.profile (or ~/.bashrc or ~/.bash_profile)中。
- (1) alias _git_reset='git reset'
 - (2) alias _git_init='git init'
 - (3) alias _git_add_all='git add .'
 - (4) alias _git_add='git add'
 - (5) alias _git_status='git status'
 - (6) alias _git_difftool='git difftool'
 - (7) alias _git_difftool_name='git difftool --name-status' # old-commit new-commit
 - (8) alias _git_commit='git commit -a'
 - (9) alias _git_branch='git branch'
 - (10) alias _git_checkout='git checkout'
 - (11) alias _git_log='git log --graph --decorate'
 - (12) alias _git_log_graph='git log --graph --oneline --decorate'