

Python版编程规范

命名规范

变量名采用小写蛇形风格,

例如 `coupon_name`、`user_last_name` 等;

常量名采用大写蛇形风格,

例如 `COUPON_NAME`、`USER_LAST_NAME` 等;

函数名采用小写蛇形风格, 并要求以动词开头,

例如 `take_last_name`、`translate` 等;

类名采用大驼峰命名风格,

例如 `Storage`、`ImageDownload` 等;

接口名采用大驼峰命名风格,

例如 `StorageInterface`、`AbstractDownload` 等;

包名采用小写蛇形命名风格,

例如 `file_storage`、`components` 等;

文件名采用小写蛇形命名风格,

例如 `utils.py`、`remote_cache.conf` 等;

项目名采用小写蛇形命名风格,

例如 `sequence`、`translation_machine` 等;

类名、包名、文件名、项目名、变量名、常量名用名词或名词短语, 如用名词短语, 定语在前, 名词在后;

例如: `ticket_vending_machine`、`document_management`

函数名用动词或者动词短语, 如用动词短语, 动词在前, 名词在后; 例如: `get_index`

表示数量的词统一使用“nums_单数”的形式; 例如: `nums_layer`, `nums_channel`

长度控制在 5 个单词内, 追求见名知意; 不要占用语言或者系统保留词; 不要使用简短的字母或者非标准的单词缩写作为名称;

例如: 用`layer_idx`表示`layer`的`index`, 用`layer_id`表示`layer`的`identity`, 不要使用`lay_id`表示`layer`的`index`。

项目名、包名、文件名对应的单词可以用复数形式表示;

接口名、抽象类名、基类名、异常类名、错误类名以特定的单词开头或结尾,

例如 `TimeException`、`StorageBase`、`PermissionInterface`、`ParseError` 等;

名称中除了字母和下划线外, 不应包含其他字符(特殊字符和数字);

例如不要出现`a2b`这样的命名;

注释标准

行尾注释: `EXPIRE = 3600` 空格空格#空格调用方指定的时间数值

单一空行注释: 1) #空格函数添加类型注释

2) """只有 1 行内容时, 注释符号的开头和结尾在同一行"""

多行注释: """消费者

当有多行内容时, 注释符号的结尾单独一行, 且第一行文字注释紧跟在注释符号的后面

"""

类型注释:

为每个变量给出类型注释`name: str, number: int = 1003,`

`from typing import Tuple, List, Dict`

```
coupons: Tuple [str, str, int] = ("编写类型注释", "像c/java", 100)
```

函数注释: 函数需要给出形参和返回值注释

```
def send(message: str, host: str, name: str) --> str:
```

```
""" 消息推送器
```

将消息推送到指定服务器的指定队列中

```
: param message: 消息体
```

```
: param host: 服务器地址
```

```
: param name: 队列名称
```

```
: return: 推送结果
```

```
"""
```

除了让人更加能够阅读之外, 另一个好处是print(send.**annotations**)时会看到更多的信息。

文件注释: 文件注释写在文件最开头, 通常被用于表明法律或版权信息以及文件内容的相关信息

```
"""
```

```
@description: 关于文件注释的描述与示例
```

```
@其他内容: 参考orion或其他项目, 但是一般公司是不写author的名字等信息的
```

```
"""
```

代码缩进与空格的使用标准

1、缩进: tab键需要变成4个空格; 单行字符数限制在 79 个内, 不要到达 80 个字符, 如达到, 需要采用悬挂缩进, 如果悬挂缩进时有符号, 将符号放下一行的行首;

```
def reader(name: str, path: str, storage: FileStorage,
```

```
header_line: int = 1, classify: str = "csv",
```

```
threshold: int = 100) -> FileStorage:
```

```
consumer = (age
```

- height
- weight
- threshold)
- name空格=空格[
- "Java", "Python", "Golang",
- "Rust", "C++", "C",
-]

2、空格:

表达式中符号左右的对象与符号以 1 个空格进行分隔; 例如: a == b

括号内不要有空格; 例如: spam(ham[1], {eggs: 2}, [])

不要在逗号, 分号, 冒号前面加空格, 但应该在它们后面加(除了在行尾); 例如 x, y = y, x

相邻行的表达式不应按照符号进行强制对齐;

参数列表, 索引或切片的左括号前