Java命名规范

- 基本原则
- 常量命名
- 变量命名
- 类与接口命名
- 方法命名
- 包命名

基本原则

1. 选择有意义的名字，见名知意！尽量一看到名称就知道代表什么意思
2. 在无歧义的前提下，命名力求简洁
3. 在语义明了的前提下，命名力求省力，如果可以同时不按shift，则不按
4. 一个标识符最好一个单词，尽量不要超过3个单词，每个单词尽量不要超过10个字符
5. 单词应当拼写正确完整
6. 拼音与英文单词不能混用

常量命名

- 常量是在作用域内保持不变的值，一般使用final进行修饰。
- 一般分为三种（按常量出现的位置）
  - 全局常量（public static final修饰）
  - 类内常量（private static final 修饰）以及
  - 局部常量（方法内，或者参数中的常量）

常量命名

- 规则
  - 全局常量和类内常量全部用大写字母表示，如果名字必须用多个单词，用下划线“_”连接
  - 常量一般不建议用缩写
  - 无论什么时候，均提倡应用常量取代数字、固定字符串。也就是说，程序中除0，1以外，尽量不应该出现其他数字（字面量）

- 常量命名实例
  - 全局常量(正例)
    - public static final long MAX_VALUE = 3600;
  - 类内常量
    - private static final String ERROR_MESSAGE HUMAN_STATUS_OK;
  - 局部常量
    - final int HUMAN_STATUS_OK= 1;

变量命名

- 一般规则
  - 变量名如果是一个单词，则小写
  - 变量名如果是2个或两个以上，第一个单词首字母小写，从第二个单词开始首字母大写（小驼峰式命名法）

- 变量命名具体规则
  - 1. 变量命名一般的基本结构为typeVariableName，使用3字符前缀来表示数据类型，例如

数据类型或对象类型	缩写
byte	bye
short	sht
Integer/int	int
Long/long	lng
float	flt
Double/double	dbl
char	chr
boolean	bln
String	str
数组	在typeVariableName基础上加前缀-a，如字符串数组：astrStudentName
自定义对象	自定义类型的变量可以采用本身的名称，把首字母改为小写

- 变量命名具体规则
  - 2. 除非是在循环中，否则一般不推荐使用单个字母作为变量名
    - i、j、k等只作为小型循环的循环索引变量
    - 个别临时变量
      - 通常被取名为i，j，k，m 和n，它们一般用于整型；
      - c，d，e，它们一般用于字符型

- 变量命名具体规则
  - 3. 如果需要对变量名进行缩写时，一定要注意整个代码中缩写规则的一致性。
    - 例如，如果在代码的某些区域中使用intCnt，而在另一些区域中又使用intCount，就会给代码增加不必要的复杂性。
    - 建议变量名中尽量不要出现缩写

- 变量命名具体规则
  - 4. 在同一方法中尽量不使用同一个变量表示前后意义不同的两个数值
  - 5. 逻辑变量：避免用flag来命名状态变量，用is来命名逻辑变量，例如

```
if(isClosed){
    dosomeworks;
    return;
}
```

- 变量命名具体规则
  - 6. 成组变量的命名，通过在结尾处放置一个量词，可创建更加统一的变量，以便于理解和搜索。例如，变量可命名如下：
    - 请使用 strCustomerFirst、strCustomerLast
      - 注意不使用strFirstCustomer、strLastCustomer。
    - 常用的量词后缀有：
      - First（一组变量中的第一个）
      - Last（一组变量中的最后一个）
      - Next（一组变量中的下一个变量）
      - Prev（一组变量中的上 一个）
      - Cur（一组变量中的当前变量）。

类与接口命名

- 类名和接口名使用大驼峰命名形式，即首字母大写原则，类的名字必须由大写字母开头而单词中的其他字母均为小写；如果一个类名称是由多个单词组成，则每个单词的首字母均应为大写，例如：ModelAboutAction;
- 类名通常使用名词或名词短语，接口名除了用名词和名词短语以外，还可以使用形容词或形容词短语，如Cloneable，Callable等，表示实现该接口的类有某种功能或能力。
- 对于一些特殊特有名词缩写也可以使用全大写命名，例如，XMLHttpRequest

类或接口	命名方式	示例
抽象类	Abstract 或Base 为开头	BaseUserService
工具类	Utils作为后缀	StringUtils
异常类	Exception结尾	RuntimeException
接口实现类	接口名+ Impl	UserServiceImpl
测试类	Test为后缀	UserServiceTest，表示用来测试UserService类的

方法命名

- 方法命名采用小驼峰命名的形式，首字小写，往后的每个单词首字母都要大写
- 和类名不同的是，方法命名一般为动词或动词短语，与参数或参数名共同组成动宾短语，即动词＋名词。
- 一个好的方法名一般能通过名字直接获知该函数实现什么样的功能。
- 以下给出常用的方法命名规范

返回真伪值的方法

前缀单词	意义	示例
is	对象是否符合期待的状态	isValid
can	对象能否执行所期待的动作	canRemove
should	调用方执行某个命令或方法是好还是不好、应不应该或者推荐还是不推荐	shouldMigrate
has	对象是否持有、所期待的数据和属性	hasObservers
needs	调用方是否需要、执行某个命令或方法	needsMigrate

与数据相关的方法

单词	意义	示例
create	新创建	createAccount
new	新创建	newAccount
from	从既有的某物新建或是从其他的数据新建	fromConfig
to	转换	toString
update	更新既有某物	updateAccount
load	读取	loadAccount
fetch	远程读取	fetchAccount
delete	删除	deleteAccount
remove	删除	removeAccount
save	保存	saveAccount
store	保存	storeAccount
commit	保存	commitChange
apply	保存或应用	applyChange
clear	清除或是恢复到初始状态	clearAll
reset	清除或是恢复到初始状态	resetAll

与集合操作相关的方法

单词	意义	例
contains	是包含指定对象相同的对象	contains
add	添加	addJob
append	添加	appendJob
insert	插入到下标n	insertJob
put	添加与key对应的元素	putJob
remove	移除元素	removeJob
enqueue	添加到队列的最末位	enqueueJob
dequeue	从队列中头部取出并移除	dequeueJob
push	添加到栈头	pushJob
pop	从栈头取出并移除	popJob
peek	从栈头取出但不移除	peekJob
find	寻找符合条件的某物	findById

成对出现的动词（一）

单词	意义
get获取	set 设置
add 增加	remove 删除
create 创建	destory 移除
start 启动	stop 停止
open 打开	close 关闭
read 读取	write 写入
load 载入	save 保存
create 创建	destroy 销毁
begin 开始	end 结束
backup 备份	restore 恢复
import 导入	export 导出
split 分割	merge 合并
inject 注入	extract 提取
attach 附着	detach 脱离
bind 绑定	separate 分离
view 查看	browse 浏览
edit 编辑	modify 修改
select 选取	mark 标记
copy 复制	paste 粘贴

成对出现的动词（二）

undo 撤销	redo 重做
insert 插入	delete 移除
add 加入	append 添加
clean 清理	clear 清除
index 索引	sort 排序
find 查找	search 搜索
increase 增加	decrease 减少
play 播放	pause 暂停
launch 启动	run 运行
compile 编译	execute 执行
debug 调试	trace 跟踪
observe 观察	listen 监听
build 构建	publish 发布
input 输入	output 输出
encode 编码	decode 解码
encrypt 加密	decrypt 解密

成对出现的动词（三）

compress 压缩	decompress 解压缩
pack 打包	unpack 解包
parse 解析	emit 生成
connect 连接	disconnect 断开
send 发送	receive 接收
download 下载	upload 上传
refresh 刷新	synchronize 同步
update 更新	revert 复原
lock 锁定	unlock 解锁
check out 签出	check in 签入
submit 提交	commit 交付
push 推	pull 拉
expand 展开	collapse 折叠
begin 起始	end 结束
start 开始	finish 完成
enter 进入	exit 退出
abort 放弃	quit 离开
obsolete 废弃	depreciate 废旧
collect 收集	aggregate 聚集

包命名

- 包名统一使用小写，点分隔符之间有且仅有一个自然语义的英文单词或者多个单词自然连接到一块（如 springframework，deepspace不需要使用任何分割）
  - 包名按照域名的范围从大到小逐步列出，恰好和Internet上的域名命名规则相反
- 包名统一使用单数形式，如果类名有复数含义，则可以使用复数形式。
- 包名的构成可以分为以下几四部分【前缀】【发起者名】【项目名】【模块名】。
  - 常见的前缀可以分为以下几种：

前缀	例	含义
indi 或 onem	indi.发起者名.项目名.模块名……	个体项目，个人发起，但非自己独自完成，可公开或私有项目，copyright主要属于发起者。
pers	pers.个人名.项目名.模块名……	个人项目，指个人发起，独自完成，可分享的项目，copyright主要属于个人
priv	priv.个人名.项目名.模块名……	私有项目，指个人发起，独自完成，非公开的私人使用的项目，copyright属于个人。
team	team.团队名.项目名.模块名……	团队项目，指由团队发起并由该团队开发的项目，copyright属于该团队所有
顶级域名	com.公司名.项目名.模块名……	公司项目，copyright由项目发起的公司所有

- 例如，我们的作业包的命名

cn.pku.javaprogramming.homework1.bean

END