Atitit 高效率开发规范

|  |  |  |  |
| --- | --- | --- | --- |
| 1.0.0 | 技术部 | 2019.12.7 | 首次向Java业界公开 |

目录

[1. 命名规约 1](#_Toc31546)

[1.1. 推荐 1](#_Toc32375)

[1.2. 不推荐 2](#_Toc5951)

[2. (三) 格式规约 2](#_Toc32606)

[3. (四) OOP规约 2](#_Toc8651)

[3.1. 推荐 2](#_Toc5010)

[3.2. 不推荐，应该避免的地方 2](#_Toc14021)

[4. (六) 并发处理 2](#_Toc6275)

[4.1. 推荐 2](#_Toc12918)

[5. (八) 注释规约 2](#_Toc904)

[6. 二、异常日志 3](#_Toc24130)

[6.1. (一) 异常处理 3](#_Toc9271)

[7. MYSQL规约 3](#_Toc20512)

[8. 四、工程规约 3](#_Toc4729)

[8.1. (一) 应用分层 3](#_Toc31456)

[9. 三、单元测试 3](#_Toc28228)

[10. ref 4](#_Toc7547)

# 命名规约

## 推荐

命名可以以美元符号开始，方便与php等语言兼容，并方便识别变量等，提升士气，到处都是dollor

使用中英文混合，也可以使用纯中文命名，提升理解度，大量的减少了无畏注释。。相当于强制提升可读性

如果某些情况对纯中文支持不好，可以使用拼音代替，注意分词

正确的拼写和语法可以让阅读者易于理解，避免歧

全面提升可读性，中文可以使用在java类名，属性，数据库库名表名，字段,sp,udf等各个地方

## 不推荐

复杂单词不要使用纯英文

# (三) 格式规约

默认即可

# (四) OOP规约

## 推荐

直接访问属性，不要通过get set，麻烦，语法噪音、

## 不推荐，应该避免的地方

少使用DO/DTO/VO等POJO类时

# (六) 并发处理

## 推荐

可以多使用数据库提供的unique索引，trigger并发机制，提升开发效率

其次使用编程语言的并发控制机制

使用TPL模式任务并发库模式，少使用线程模式这种底层模式

# (八) 注释规约

直接使用中文或附加中文的方式来命名，避免过多的注释

【推荐】与其“半吊子”英文来注释，不如用中文注释把问题说清楚。专有名词、关键字，保持英文原文即可

# **二、异常日志**

## **(一)异常处理**

**推荐使用lomok来消除异常警告**

优先异常用来做流程控制，条件控制，因为可读性更高。

优先使用异常控制，避免使用返回码（包括http接口）

# **MYSQL规约**

【强制】禁用保留字，如desc、range、match、delayed等，参考官方保留字

【推荐】单表行数超过500万行以上或更高根据情况而定，可以优先进行分区

或者单表容量超过2GB，才推荐进行分区分库分表。

推荐使用唯一索引，尽可能避免在编码在应用层通过“先查后插”

优先使用使用外键与级联，避免在在应用层解决。

可以使用存储过程，存储过程无需编译部署，调试也可以

推荐直接拿HashMap与HashTable作为查询结果集的输出。不要使用bean实体转换，繁琐还消耗性能。。

# **四、工程规约**

## **(一)应用分层**

简化分层，避免过度设计

少使用领域模型，多使用db模型

# **三、单元测试**

# ref

阿里巴巴Java开发规范（终极版v1.3.0）--个人整理 - 呢喃北上的博客 - CSDN博客.html

阿里巴巴Java开发规范手册 - HarkLee - 博客园.html

阿里巴巴Java开发规范手册 - HarkLee - 博客园.html