|  |  |  |
| --- | --- | --- |
| Java代码评审检查单 | | |
| 检查项 | 描述 | 状态 |
| 检查命名规范 | 包名、类名、接口名、抽象类、缺省实现、实现类名Utility Classes、 变量命名是否符合规范? |  |
| 方法名称是否符合规范? |  |
| 检查变量声明顺序 | 类变量声明顺序是否符合规范? |  |
| 方法变量声明顺序是否符合规范? |  |
| 检查控制相关的规范 | 是否避免了深度嵌套? |  |
| 变量是否初始化? |  |
| 代码是否是可测试的? |  |
| 函数据输入输出是否清楚? |  |
| 是否有未初始化的循环和条件变量? |  |
| 代码是否容易重用和易于测试? |  |
| 检查清楚易懂性 | 代码易读并有良好的注释来理解代码? |  |
| 是否有修改历史以反映变更? |  |
| 在嵌套的最后语句，嵌套是否终止并注释指出控制终止？ |  |
| 使用的缩进是否一致？ |  |
| 不必要的代码是否已经删除? |  |
| 编码格式是否符合规范? |  |
| 是否为每块提供注释?注释是否符合规范? |  |
| 检查模块化规范 | 模块大小是否理想? |  |
| 模块是否有负作用？ |  |
| 模块/类要包含如下明细:   1. 产品名 2. 模块名 3. 变更请求标识 4. 作者名 5. 其它 |  |
| 检查错误处理规范 | 为例外条件是否做了错误处理? |  |
| 在程序异常时，是否有有用信息提示或温和地退出? |  |
| 是否检测和控制错误状态? |  |
| 检查普通规范 | 代码与详细设计是否匹配? |  |
| 硬编码值在代码中是否避免，是否可应用? |  |
| 接口描述是否包括一 个调用序列样板并对所有参数进行描述? |  |
| 代码的调试版式本和发布版本是否被维护? |  |
| 检查可能的最大最小值 |  |
| 检查越界下标 |  |
| 检查错误状态 |  |