**JAVA代码评审标准清单（Code Checklist ）**

|  |  |  |
| --- | --- | --- |
| 常规 | 是/ 否 | 备注 |
| 程序逻辑是否符合详细设计 |  |  |
| 程序是否明确地注释了其实现功能 |  |  |
| 注释格式是否正确 |  |  |
| 程序单元是否承担单一职责 |  |  |
| 复杂程序是否合理地分解为子程序 |  |  |
| 程序中接口定义是否明晰 |  |  |
| 异常处理是否符合规范 |  |  |
| 包名、类名命名是否符合规范 |  |  |
| 方法名命名是否符合规范 |  |  |
| 代码中段落布局是否合理、是否用恰当的空行 分隔 |  |  |
| 代码的断行、对齐、缩进、空行是否恰当 |  |  |
| 代码中的变量、属性、参数、方法、类是否恰当命名，是否无相互混淆的命名 |  |  |
| 其他编码风格是否符合规范要求，如if、else永远加{}；for、while的格式；switch永远有default；case有break等等基本语句格式 |  |  |
| 记录错误日志使用架构统一方法 |  |  |
| 下标的值在范围之内且不存在非整数下标 |  |  |

|  |  |  |
| --- | --- | --- |
| 变量 /常量 | 是/ 否 | 备注 |
| 相关系列变量是否统一集中声明，自定义数据类型变量是否进行了必要而充分的描述 |  |  |
| 变量是否按相关规范进行了命名（有意义的命名、数据类型的区分） |  |  |
| 变量是否按其命名含义承担了单一的使用目 的，必要时是否清晰定义了额外变量 |  |  |
| 字段是否使用 getter/setter方法访问 |  |  |
| 是否应用良好定义的枚举类型替代了简单的字 符标识或布尔变量使用 |  |  |
| 处理可变 String时，是否使用 StringBuffer/ StringBuilder |  |  |
| 是否优先使用 Java标准库提供的容器 （ArrayList 、HashSet、HashMap、LinkList ）来 处理顺序结构、集合、关联数组合堆栈队列 |  |  |
| 成员变量和方法 public/private/protected 关 键字是否恰当 |  |  |
| 内部类、非公共类及其内部成 员、方法的访问权限是否合适 |  |  |
| 不同作用域变量名称尽量不要相同 |  |  |

|  |  |  |
| --- | --- | --- |
| 算法 | 是/ 否 | 备注 |
| 数据结构是否合理、精简 |  |  |
| 算法是否可以独立测试，是否与数据库和其他 算法隔离 |  |  |
| 是否尽量避免了嵌套的运用 |  |  |
| 复杂逻辑是否进行了必要而充分的注释 |  |  |

|  |  |  |
| --- | --- | --- |
| 控制 | 是/ 否 | 备注 |
| 代码执行路径是否清晰 |  |  |
| 控制逻辑复杂度是否合理，是否进行了必要而 充分的注释 |  |  |
| 每个循环体是否仅执行了单一而明确的功能 |  |  |
| 与常数比较需要将常数放在比较表达式的前面 |  |  |
| 没有使用任何实例类成员（包括方法和成员变 量）的方法是否被声明为静态的 |  |  |

|  |  |  |
| --- | --- | --- |
| 布局 | 是/ 否 | 备注 |
| 代码是否良好格式化并能体现其逻辑结构 |  |  |

|  |  |  |
| --- | --- | --- |
| 设计 | 是/ 否 | 备注 |
| 程序是否可读、可扩展并健壮 |  |  |
| 具体实现细节是否已尽可能的 “隐藏 ” |  |  |
| 是否优先使用接口而不是抽象类或具体类 |  |  |
| 方法参数是否在 5 个以内 |  |  |
| 方法慎用 public 修饰符 |  |  |
| 对于不需要子类来重载的类尽量使用 final |  |  |
| 对象作用域是否过大 |  |  |

|  |  |  |
| --- | --- | --- |
| 效率 | 是/ 否 | 备注 |
| 尽量不要在循环内出现远程调用 |  |  |
| 每个业务动作远程调用次数是否小于 3 次 |  |  |
| 远程调用数据传输是否有不必要的冗余数据 |  |  |
| 如果已知道数据长度， new一个 List 或Map时， 加上长度参数 |  |  |

|  |  |  |
| --- | --- | --- |
| 输入流 /输出流检查 | 是/ 否 | 备注 |
| 文件属性正确 |  |  |
| I/O语句符合格式规范 |  |  |
| 缓冲大小与记录大小相匹配 |  |  |
| 文件在使用后是否关闭 |  |  |
| 是否处理了 I/O错误 |  |  |

|  |  |  |
| --- | --- | --- |
| SQL 检查 | 是/ 否 | 备注 |
| Sql语句小写 |  |  |
| 引用字符使用单引号 |  |  |
| 严禁使用 select \* 形式的语句，必须指出具体字段 |  |  |
| 严禁使用 insert into table values（？，？，？），必须指出具体要赋值的 字段 |  |  |
| 避免隐含的类型转换（不同数据类型字段相 加） |  |  |
| 子查询前后必须加上括号 |  |  |
| 禁止使用视图 |  |  |
| 禁止使用 XX in () or XX in () |  |  |
| 禁止使用 or超过 500个 |  |  |
| 禁止使用 not in，建议使用 not exist |  |  |
| 禁止在一条 sql语句中使用 3层以上的嵌套 |  |  |
| 如果有多表连接时，应该有主从之分，尽量从 一个表取数 |  |  |
| Where子句过滤条件，索引列或过滤记录最多 的条件应该放在前面 |  |  |
| 字符串连接必须使用 “|| ” |  |  |
| Case when语句中只能出现 =、>=、<=以及 is null 运算符 |  |  |
| Sql 中函数是否在指定范围内 |  |  |
| Join 与 on 必须严格匹配 |  |  |

|  |  |  |
| --- | --- | --- |
| 目录结构 | 是/ 否 | 备注 |
| Jsp,Action ，service ，dao , 逻辑分层是否明 确 |  |  |
| 配置文件（ struts ，spring ，sql ）是否条理清 晰，目录规范 |  |  |
| 页面资源文件（ JavaScript,CSS,images ）目录 是否规范 |  |  |
|  | | |
| 其他检查 |  |  |
| 不存在“警告”或“提示”信息 |  |  |
| 是否有输入信息的合法性检查 |  |  |
| 是否有功能遗漏 |  |  |