

代码注释是架起程序设计者与程序阅读者之间的通信桥梁，最大限度的提高团队开发合作效率。也是程序代码可维护性的重要环节之一。所以我们不是为写注释而写注释。下面说一下我们在诉求网二期开发中使用的代码注释规范，供大家参考下。

原则：

1、注释形式统一

在整个应用程序中，使用具有一致的标点和结构的样式来构造注释。如果在其它项目中发现它们的注释规范与这份文档不同，按照这份规范写代码，不要试图在既成的规范系统中引入新的规范。

2、注释内容准确简洁

内容要简单、明了、含义准确，防止注释的多义性，错误的注释不但无益反而有害。

注释条件：

1、基本注释（必须加）

(a) 类（接口）的注释

(b) 构造函数的注释

(c) 方法的注释

(d) 全局变量的注释

(e) 字段/属性的注释

备注：简单的代码做简单注释，注释内容不大于10个字即可，另外，持久化对象或VO对象的getter、setter方法不需加注释。具体的注释格式请参考下面举例。

2、特殊必加注释（必须加）

(a) 典型算法必须有注释。

(b) 在代码不明晰处必须有注释。

(c) 在代码修改处加上修改标识的注释。

(d) 在循环和逻辑分支组成的代码中加注释。

(e) 为他人提供的接口必须加详细注释。

备注：此类注释格式暂无举例。具体的注释格式自行定义，要求注释内容准确简洁。

注释格式：

1、单行(single-line)注释：`//....."`

2、块(block)注释：`/*.....*/"`

3、文档注释：`/**.....*/"`

4、javadoc 注释标签语法

@author 对类的说明 标明开发该类模块的作者

@version 对类的说明 标明该类模块的版本

@see 对类、属性、方法的说明 参考转向，也就是相关主题

@param 对方法的说明 对方法中某参数的说明

@return 对方法的说明 对方法返回值的说明

@exception 对方法的说明 对方法可能抛出的异常进行说明