Easy Opensource项目编码规范

# 标示符命名

标示符命名符合一般java程序的编码规范，参考以下表格：

|  |  |  |
| --- | --- | --- |
| 标示符类型 | 命名规则 | 例子 |
| Packages | 每一个包的名称总是小写，规则采用类似于TCP/IP中域名的反序，即以com, edu, gov, mil, net, org, 或ISO 3166,1981中定义的两位国家代码开始，接着是组织名称，在此处以com.topgroup开始，接着是<项目英文简写>，其后的部分由项目自已定义。 | com.topgroup.eng  com.topgroup.report.v2  com.topgroup.jdm |
| Classes | 类名必须是名词，大小写遵照前面定义的大小写规则（即每个单词首字母大写）。类名应该简单清晰。不要使用除通用的计算机缩写或通用的领域缩写外的其它缩写。 | class Raster; class ImageSprite; |
| Interfaces | 同类名规则。 | interface RasterDelegate; interface Storing; |
| Methods | 方法应该是动词，或以动词开始的动宾结构短语，大小写遵照前面定义的大小写规则(即除第一个单词外，每个单词的首字母大写)。 | run(); runFast(); getBackground(); |
| Variables | 变量名应该短而准确并便于记忆。大小写遵照前面定义的大小写规则(即除第一个单词外，每个单词的首字母大写)。除了临时使用的如循环变量等以外，不要使用单字符的变量名。对于使用单字符的临时变量名，建议循环变量名用i,j,k，n表示整数，c,d,e表示字符。 | int i;  char c;  float myWidth; |
| Constants | 全部大写，单词之间用下划线分隔。 | static final int MIN\_WIDTH = 4;  static final int MAX\_WIDTH = 999;  static final int GET\_THE\_CPU = 1; |

# 注释要求

请在每次编写代码之前考虑是否已补全注释

以下两种注释必须给出

## 类注释

每个类声明之前必须加上以下注释

注释格式

/\*\*

\*类功能说明：

\* @author 张灵箫（类的作者写上自己的名字）

\*/

## 方法注释

方法体只有一到两行的，功能十分明确的方法（如getter，setter）可以不加注释

除此之外，功能需要说明的，方法体达到一定长度的（5行以上）方法之前必须加上以下注释

/\*\*

\*方法功能说明：

\*@author 张灵箫（函数的作者写上自己的名字）

\*修改者 修改日期

\*修改内容 （如果非原作者对方法有修改，请在修改前加上以上注释）

\*/

## 其他注释

在方法体内对复杂的算法和使用方法需要说明的变量需要给出方法内的注释

可以用单行或者多行注释。

# 异常处理

本项目自上而下按照顺序有：

* Servlet层
* Business/crawler层
* 数据接口层

异常处理遵循以下原则：

1. 下层只向上层抛出异常，例如数据接口层只能向servlet层抛出异常
2. 所有异常在Business/crawler层处理，Business/crawler层不能向servlet层抛异常，必须catch到所有异常并做相应处理。而数据接口层出现异常，则直接抛出，无需处理。
3. Crawler也可以做异常处理，crawler不要向business抛异常，而应该按异常类型将错误值返回给business层