# 1 Java命名规范

## 1.1通用命名规则

### 1.1.1 Pascal命名

Pascal:单个或由多个英文单词连接，每个单词首字母大写，

例：HomeController、StringHelper、ProjectDao

### 1.1.2 Camel命名

Camel:单个或由多个英文单词连接，首个单词的首字母小写，其余单词的首字母大写，

例：type、setType、getType

### 1.1.3 下划线命名

采用小写形式[必须]，多个单词使用英文状态下的下划线（\_）连接[必须]，

例：type、user\_name

### 1.1.3 简写

关于简写的问题，尽可能见文知义，如果需要简写（通常不建议使用），也要是通用的简写方式，大家都能看得懂的，

例： Aspect Oriented Programming简写为AOP

## 1.2项目、包命名

### 1.2.1项目命名

根据合同约定的系统名称的英文简写大写或者拼音的首字母小写，例：新华学生收费管理系统（xinhua student charge manage system） xhscms

### 1.2.2 包命名

包名称全部采用小写形式，以com.rz开头，后跟项目名称简写的小写形式，最后是功能模块或者类别名称

例：com.rz.cms.controller、com.rz.cms.service

## 1.3类型、类型成员

### 1.3.1 类、接口命名

采用Pascal命名规则[必须]，名词或者名词短语，且能标识事物的特性，

例：User、UserService、UserServiceImpl 、UserController

### 1.3.2 字段、常量字段命名

字段采用下划线或者Camel命名规则[必须]，名词或者名词短语，且能标识事物的特性，

例：name、user\_name、userName

常量字段采用全部大写，单词间以下划线隔开的形式，

例：LOCK\_SECONDS、RETURN\_INFO\_KEY

### 1.3.3 方法、方法参数命名

方法名采用Camel命名规则[必须]，动词或动词短语，

例：submit、submitOrder、submitBookOrder；

方法参数采用下划线或者Camel命名规则[必须]，

例：name、user\_name、userName

### 1.3.4 局部变量命名

采用下划线或者Camel命名规则[必须]，

例：name、user\_name、userName

### 1.3.5 接口成员命名

对于属性：采用Camel命名规则[必须]，名词或者名词短语，

例：getId、setId、findById

对于方法：方法名采用Pascal命名规则[必须]，动词或动词短语，方法参数采用下划线或者Camel命名规则[必须]；

例：add(User user)、findById(Guid userId) 、findById(Guid user\_id)

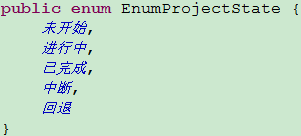
### 1.3.6 泛型参数命名

采用Pascal命名规则[必须]，名词或者名词短语，一般以大写字母T作为前缀，

例：TAggregateRoot、TKey、 TDto

### 1.3.7 枚举类型命名

采用Pascal命名规则[必须]，名词或者名词短语，以Enum开头，枚举名称能清楚的表达该枚举的用途，枚举成员可以使用汉字，



## 1.4特定于框架的相关命名

### 1.4.1 Domain、DAO层各成员命名

a.实体类命名

采用Pascal命名规则[必须] ，名词或者名词短语，且能标识实体的特性，

例：Project、UserInfo

b.DAO接口命名

实体类名+Dao

例：ProjectDao、UserInfoDao

b.DAO实现类命名

实体类名+DaoImpl

例：ProjectDaoImpl、UserInfoDaoImpl

如果使用第三方ORM框架，也可以加上表明特定框架的命名，如下使用MyBatis框架的命名

例：ProjectDaoSqlMapImpl、UserInfoDaoSqlMapImpl

### 1.4.2 Service层各成员命名

a.Service接口命名

采用Pascal命名规则[必须] ，名词或者名词短语，后面加Service，

例：ProjectService、UserInfoService

b.Service接口实现类命名

采用Pascal命名规则[必须] ，名词或者名词短语，后面加ServiceImpl，

例：ProjectServiceImpl、UserInfoServiceImpl

### 1.4.3 Web层各成员命名

a.控制器（Controller）类命名

采用Pascal命名规则[必须] ，名词或者名词短语，后面加Controller，

例：UserController、ProductTypeController

b.控制器方法（Action）命名

采用Pascal命名规则[必须] ，名词、动词或者名词、动词短语

例：index、indexList、create、validateName

c.SpringMVC映射路径命名

采用全小写形式[必须] ，名词、动词或者名词、动词短语

例：user/index、user/list、user/add

## 1.5数据库表、字段命名

### 1.5.1 表命名

采用下划线命名规则[必须] ，名词或者名词短语

例：user、system\_set

### 1.5.2 表字段命名

采用下划线命名规则[必须] ，名词或者名词短语

例：id、user\_name、contact\_info

# 2 js、html命名规范

## 2.1 js命名

### 2.1.1 js类命名

采用Pascal命名规则[必须]，

例：function StringBuilder(){}

### 2.1.2 js类成员（属性、方法）命名

采用Camel命名规则[必须]，

例：

StringBuilder.prototype.length = 10;

StringBuilder.prototype.add = function(){}

StringBuilder.prototype.getName = function(){}

### 2.1.3 js函数命名

采用Camel命名规则[必须]，

例：remove、removeTr

### 2.1.4 js变量（局部变量、全局变量、函数参数）命名

采用Camel命名规则[必须]，

例：index、tempArray

## 2.2 html(css类、标签Id)命名

### 2.2.1 css类命名

采用下划线命名规则[必须]，名词或者名词短语，也可以在最后加上标签名称，

例：content，item\_list，item\_list\_li

### 2.2.2 标签Id命名

采用下划线命名规则[必须]，名词或者名词短语，也可以在最后加上标签名称，

例：container或者container\_div、user\_name或者user\_name\_hidden

注意在同一Html文档中Id名称不能重复。

**注：**

**1.对于标有必须字样的规则，必须严格遵守；**

**2.所有命名，优先采用英文单词的形式，其次再考虑汉语拼音，尽量不要直接使用汉字（枚举成员除外）。**