**项目命名规范**

**（2018年1月16日）**

**数据库命名规范**

* **数据表名**

符合表实际用途的有意义英文内容，尽可能采用常见英文。

首字母大写其余字母小写。

若出现多个单词则每个单词的首字母大写。

表名称最好不要超过三个单词长度（22字母）。

表名一般为单数形式，若与系统名称冲突可采用复数形式区分。

* **字段名**

符合字段实际用途，采用有意义的常见英文做字段名。

以表名首字母大写以及列名首字母大写为前缀。

前缀若冲突可采用表名首单词的缩写形式开头，首字母仍大写。

避免使用容易混淆的列名。

* **存储过程名**

符合存储过程的实际用途，采用有意义的常见英文命名。

使用动词+名词的形式命名存储过程，每个单词首字母大写，如“GetClass”。

存储过程中尽量采用事务的方式执行。

存储过程按功能类分放。

**\*** 本项目不采用常见的p\_做存储过程前缀。

**编程语言命名规范**

* **变量名**

Camel 规则(大驼峰和小驼峰命名)。

除了第一个单词外的其他单词的开头字母大写. 如”myName”。

在简单的循环语句中计数器变量使用 i, j, k, l, m, n。

命名要与实际意义与用途相同。

* **方法名**

Pascal 规则（帕斯卡命名）。

每个单词开头的字母大写，如“CutData”。

对方法名采用一致的动词/宾语或宾语/动词顺序。

不要在方法中重复类的名称”。

命名要与实际意义与用途相同。

* **属性名（实体类属性）**

名称应该为名词及名词短语。

使用Pascal规则，每个单词开头的字母大写，如“MyName”。

对于bool型属性或者变量使用Is作为前缀，不要使用Flag后缀。

命名要与实际意义与用途相同。

* **类名**

类名应该为名词及名词短语，尽可能使用完整的词。

使用Pascal规则，每个单词开头的字母大写，如“StudentInfo”。

在适当的地方，使用复合单词命名派生的类。派生类名称的第二个部分应当是基类的名称。

命名要与实际意义与用途相同。

* **枚举名**

对于 Enum 类型和值名称使用 Pascal 规则。

少用缩写。

不要在 Enum 类型名称上使用 Enum 后缀。

对大多数 Enum 类型使用单数名称，但是对作为位域的 Enum 类型使用复数名称。

总是将 FlagsAttribute 添加到位域 Enum 类型。

* **集合名**

名称应该为名词及名词短语。

使用Pascal规则。

名称后面追加“Collection”。

* **接口名**

接口名称应该为名词及名词短语或者描述其行为的形容词，尽可能使用完整的词。

* **事件名**

event handlers命名使用 EventHandler 后缀。

两个参数分别使用 sender 及 e。

使用Pascal规则。

事件参数使用EventArgs 后缀。

事件命名使用语法时态反映其激发的状态，例如 Changed，Changing。

考虑使用动词命名. 变量命名。

* **其他编码规范**

代码的缩进。要用Tab，而不要用space。

局部变量的名称要有意义。不要用x，y，z等等（除用于For循环变量中可使用i，j，k，l，m，n）。

所有的成员变量声明在类的顶端，用一个换行把它和方法分开。

用有意义的名字命名namespace，如：产品名、公司名。

把相似的内容放在一起，比如数据成员、属性、方法、事件等，并适当的使用#region…#endregion。

**前端命名规范**

* **ID、Class命名**

所有的命名最好都小写。多个单词可用“-”连接

属性的值一定要用双引号("")括起来，且一定要有值如class="divcss5",id="divcss5"。

表现与结构完全分离，代码中不涉及任何的表现元素，如style、font、bgColor、border等。  
使用有意义的英文命名原则。

尽量不缩写，除非一看就明白的单词。  
类名的第一个字符不能使用数字。

* **JS变量、方法命名**

**变量**

小驼峰式命名法，除了第一个单词外的其他单词的开头字母大写，如”myName”。

前缀应当是名词。

尽量在变量名字中体现所属类型，如:length、count等表示数字类型；而包含name、title表示为字符串类型。

**函数（方法）**

小驼峰式命名法，除了第一个单词外的其他单词的开头字母大写，如”testPost”。

前缀应当为动词。

可使用常见动词约定：

|  |  |  |
| --- | --- | --- |
| **动词** | **含义** | **返回值** |
| can | 判断是否可执行某个动作(权限) | 函数返回一个布尔值。true：可执行；false：不可执行 |
| has | 判断是否含有某个值 | 函数返回一个布尔值。true：含有此值；false：不含有此值 |
| is | 判断是否为某个值 | 函数返回一个布尔值。true：为某个值；false：不为某个值 |
| get | 获取某个值 | 函数返回一个非布尔值 |
| set | 设置某个值 | 无返回值、返回是否设置成功或者返回链式对象 |
| load | 加载某些数据 | 无返回值或者返回是否加载完成的结果 |