### 关于606实验室c#代码规范

### 说明

[1. 实验室代码管理平台为Github，](mailto:1.实验室代码管理平台为Github，账号514554005@qq.com)

[账号514554005@qq.com](mailto:1.实验室代码管理平台为Github，账号514554005@qq.com) 密码199512198921xqq

1. 修改的代码于工作日晚进行同步更新；修改格式为姓名缩写加日期，若没有修改代码无需更新

示例：xxq1105

1. 关于Github安装使用方法以及代码同步更新教程可参照：

<https://note.youdao.com/share/?token=87BA4DCE2B734430ABFB18DB0919F614&gid=66836189>

<https://blog.csdn.net/cynophile/article/details/78971145>

1. 关于书写代码的基本要求

（1）避免函数过长。如果函数的代码超过300～400行，必须考虑将代码分解成小函数。

（2）避免使用常量，不使用固定值，尽可能使用变量，并注释其定义。

（3）在定义变量、方法或控件名称时，应避免使用过于简单的缩写或者单个字母，名称应尽可能显而易见地表示出其功能，以减少不必要的文档解释。

### 二、关于注释

**1.文件头部注释**  
在代码文件的头部进行注释，标注出创始人、创始时间、修改人、修改时间、代码的功能等，尽可能精短明了。   
样本：   
/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\* 作者： 席孝倩  
\*\* 创始时间： xxx  
\*\* 修改人：TOM   
\*\* 修改时间：  
\*\* 修改人：Ken   
\*\* 修改时间：  
\*\* 描述：   
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**2.函数、属性、类等注释 （可扩充）:**具体格式以*IDE* 编辑器输入*“///”* 自动生成的格式为准，说明该部分的功能作用。对类和接口的注释必须加上上述标记，对方法可以视情况考虑。

示例  
/// <summary>   
///   
/// </summary>

class ProductTypeCollector   
{  
…   
}

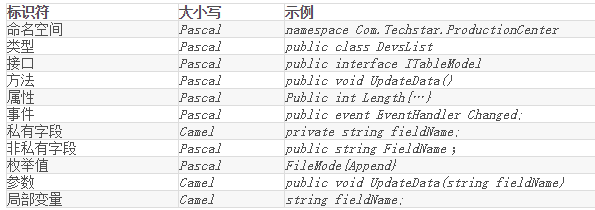
**3.单行注释：**变量的声明、代码或代码段的解释

 示例：private int number; // 注释语句

### **三、命名规范** Pascal风格 包含一个到多个单词，每个单词首字母大写，其余字母均小写。例如：HelloWorld、SetName等等。

Camel风格 包含一个到多个单词，第一个单词首字母小写，其余单词首字母大写。例如：helloWorld、productId等等。

1.关于不同命名风格的选择如下表：



1. 接口的名称加前缀 I  
   示例：  
   interface ICompare  
   {  
    int Compare();  
    }

3.自定义的异常以Exception结尾

示例：  
  public class AppException : Exception  
  {

}

4.方法的命名。一般将其命名为动宾短语。

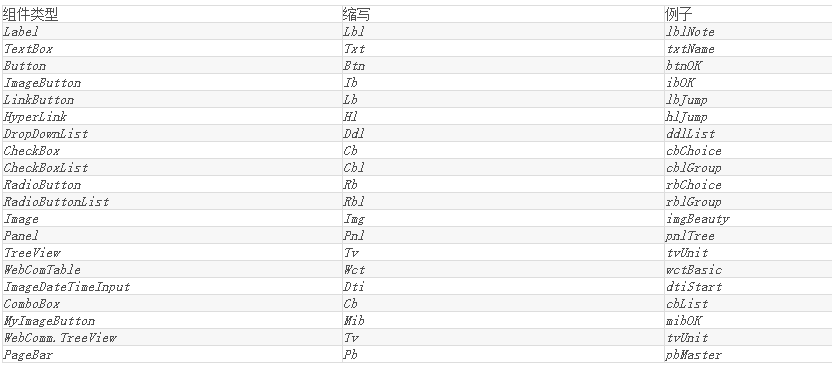
示例  
  ShowDialog()

  CreateFile()

5.控件命名规范

建议使用控件名简写作为前缀，并且简写的手字谜小写，符合Camel规范。

格式：控件名简写＋英文描述，英文描述首字母大写。



7.避免使用与常用的 .NET 框架命名空间重复的类名称。例如，不要将以下任何名称用作类名称：System、Collections、Forms 或 UI

1. **声明**  
   1. 每行声明数，一行只建议作一个声明，并按字母顺序排列。

不推荐示例： int x, y;   
2. 初始化：建议在变量声明时就对其做初始化。          
3. 变量建议置于块的开始处，不要总是在第一次使用它们的地方做声明。如  
  应避免不同层次间的变量重名，如  
4. 类和接口的声明

（1）在方法名与其后的左括号间没有任何空格。

（2）左花括号 “{” 出现在声明的下行并与之对齐，单独成行。

（3）方法间用一个空行隔开。

5.字段的声明

不要使用是public或protected 的实例字段。考虑为字段提供get和set属性访问器，而不是使它们成为公共的。带有get和set属性访问器的私有实例字段的正确使用示例：

public class Control: Component  
{  
   private int handle;  
   public  int Handle

{  
     get  
     {   
      return handle;   
     }  
   }  
}