## C#相关

### 命名相关

1. 类型、方法名、枚举用Pascal规则来命名，例如：
   1. public class PlayerInfo {}
   2. public string GetPath() {}
2. 局部变量（private,protected）、成员变量、方法的参数、枚举项用驼峰命名规则，例如：
   1. private string userName;
   2. private int userId;
   3. private int GetIdByName(string name) {}

尽量避免使用无意义的局部变量名称，比如x,y,z,循环中可以使用i,j,k…

1. 成员变量申明，请不要以“m\_”、“g\_”、“s\_”开头，同时不带类型前缀，不使用匈牙利命名法，尽量少的使用“\_”开头命名，更多的依赖编辑器，同时关注更多代码部分，例如
   1. private string name;
2. 公有成员变量和方法用Pascal规则命名，例如：
   1. public int UserId;
   2. pubic readonly string Name;
   3. pubic string GetName() {}
3. 接口名称前加前缀“I”，抽象类名称前加前缀“Abs”，例如：
   1. public interface IProxy;
   2. public abstract class AbsProtocol {}
4. 常量要用大写，多个单词之间用”\_”隔开，例如：

public const string NAME = “Name”;

public const string PASSIVE\_SKILL = “PassiveSkill”;

### 定义相关

1. 一个类比较复杂的时候可以用partial来定义类，例如把Player类拆分成Player.Data.cs, Player.Skill.cs, Player.Model.cs文件，方便查找修改。
2. 属性定义请以C#的Property访问来定义，不通过类Java的Getter或Setter方式来访问，例如：
   1. public string UserName;
   2. public readonly string UserName;
   3. public string UserName { get { return “MyName”;}, set}
   4. ~~public string GetName() { // 不允许~~

~~return “MyName”;~~

~~}~~

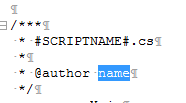
~~public void SetName(string name) {}~~

### 设置相关

1. 代码缩进，使用4个空格替代Tab。
2. cs文件、配置文件等等文本文件，统一使用UTF-8编码，VS下请使用修改过的模板创建源文件。



请把上面三个文件替换unity的模板文件（C:\Program Files\Unity\Editor\Data\Resources\ScriptTemplates），并把作者名字改成自己的名字，如下图：



### 代码相关

1. 变量声明，一行只声明一个变量，例如：
   1. public string Name;
   2. private string name;

不允许出现public string Name, Age, Sex;

1. 成员变量的声明集中在类的顶端，成员变量声明区域与第一个方法开始处用一空行隔开，方法与方法中间有一空行隔开。
2. 生成和构建一个长的字符串，一定使用StringBuilder，而不是用string+。
3. 始终使用“{}”来包含if下的语句，即使是只有一条语句。
4. 根命名空间以Xsjm开始，子命名空间以功能划分，且命名对应硬盘上的文件目录，例如：
   1. namespace Xsjm.Logic {}
   2. namespace Xsjm.Core.Net {}
5. 不要使用System.Linq, 防止编译发生问题。
6. 不要使用foreach来遍历集合，使用for循环替代，减少GC。

## Unity场景相关

1. 使用空得GameObject来做场景目录并置于原点（0，0，0），例如：Models{m1,m2……},Cameras{MainCamera,PersonCamera…}(UI的camera除外)，Lights{Light1,Light2…}，场景目录的命名也遵守Pascal命名规则。
2. 尽量把资源做成Prefab并把同类的资源放在单独的目录下，方便维护。
3. 临时对象或者测试的对象用”\_\_”开头(双下划线开头)，如果已经有一个PlayerPrefab,测试的就叫\_\_PlayerPrefab.

### UI相关

1. UI中得各个功能模块名称尽量以xxxPanel结尾。例如：RechargePanel, ActivityPanel….,并且各个功能模块应该是Prefab,并且带UIPanel组件，内部如有需要也应该用panel区分，便于控制层级结构。