配置表使用说明文档

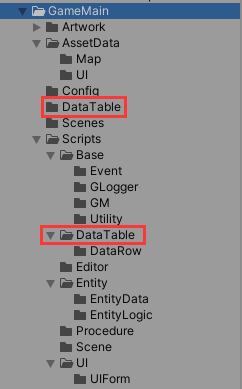
配置表的作用不仅仅是数据的保存和传递，加快游戏的开发效率和降低数据管理成本，更是对游戏设计逻辑的体现。

--Cherry

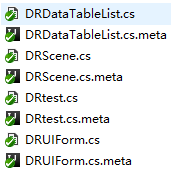
## 配置新表流程

配置表的文本文件路径在GameMain/DataTable，也是我们游戏中读取的配置表数据文件。

配置表对应的的脚本文件在DataTable/DataRow中，每个配置表文本会对应一个相应的脚本文件。



对应的配置表脚本是自动生成的，根据txt文件的名称前面有DR的前缀。

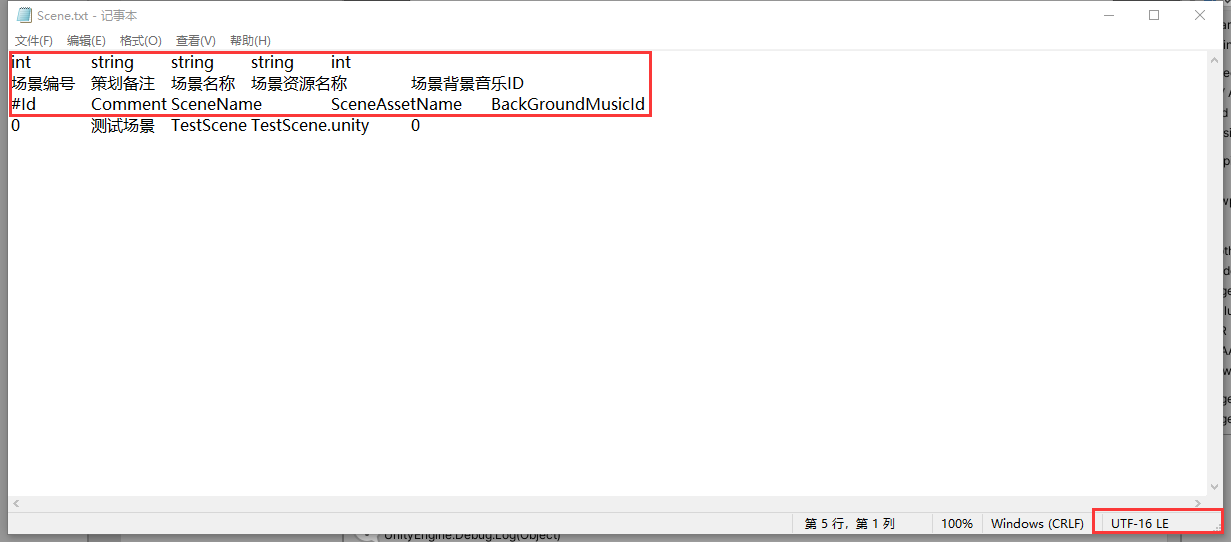
![C:\Users\zhuangyifan\AppData\Roaming\Tencent\Users\965491096\TIM\WinTemp\RichOle\QGL](ZI45WESWR{DI{~M@@B.png](data:image/png;base64,)

配置新表步骤：

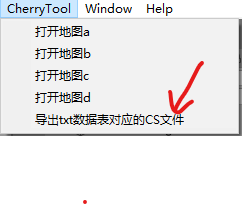
1.在GameMain/DataTable中选择之前的一个配置表进行复制重命名为新表，这样能保证表的译码格式和内容格式都是正确的。

2.创建新表后再GameMain/DataTable/DataTableList中配置对应的新表的行数据，按照之前配置的模式即可，这一步是为了游戏能对应加载后面的配置表文件。

3.现在可以开始配置新表了，以UI表为例。第一行数据是表数据参数类型，第二行数据代表数据变量的备注，第三行代表数据变量名称，注意第三行的一个字段前面需要加上#号。



4.配置好基本数据后就可以配置后面的文件了，配置完毕后要使用导表工具生成对应的脚本文件。生成的文件位置还是如上面所说。



## 配置表数据事项

目前配置表支持类型较少，后续可根据需求添加，如果是**测试的行数据可以在每行开头添加一个#号**，这样程序将不会读取该行数据。**Txt文本文件的格式必须是UTF-16-LE**，如果是新建txt文件的方式配表，请将表文件配置成该格式。生成表文件后首先看生成的代码文件是否有格式错误，如果有，那就是前三行数据配置错误，然后进入游戏进行测试如果发现数据不正确请先检查数据是否写入表中。

Cherry中配置表相关脚本文件说明：

* + DataRowCreator.cs 自动生成数据表脚本工具
  + GameDataTableHelper.cs 用于游戏中读取数据表的解析方式。
  + DataTableExtension.cs 用于扩充数据表组件功能，包括后续的扩展表数据类型解析方法。
  + DataRow等文件，表行数据脚本，与数据表文本文件对应，代表一个表的数据结构。

## 配置表目前支持的数据类型（可后续扩展）

1. 整数类型，第一行数据配置为int
2. 字符串类型，第一行数据配置为string
3. 布尔类型，第一行数据配置为int，0代表false，1代表true。
4. 待完善….