**Unity3D中C#使用StreamWriter类写入文件**

Posted on 2013年03月25日 by U3d / [Unity3D 基础教程](http://www.unitymanual.com/category/manual/unity3d-%e5%9f%ba%e7%a1%80%e6%95%99%e7%a8%8b)/被围观 301 次

[**Unity3D**](http://www.unitymanual.com)中C#使用StreamWriter类写入文件。除了使用FileStream类读写文本文件，.net还提供了StreamWriter类和StreamReader类专门处理文本文件。这两个类从底层封装了文件流，读写时不用重新编码，提供了更文件的读写方式。StreamWriter类允许将字符和字符串写入到文件中，不必转换为字节，它处理底层的转换，向FileStream对象写入数据。

相关文章：[**Unity3D中C#使用StreamReader类读取文件**](http://www.unitymanual.com/3213.html)

一.创建StreamWrite对象

如果已经有了FileStream对象，则可以使用此对象来创建StreamWriter对象：

FileStream fs = new FileStream("test.txt",FileMode.CreateNew);

StreamWriter sw = new StreamWriter(fs);

也可以直接从文件中创建StreamWriter对象：

StreamWriter sw = new StreamWriter("test.txt",true);

此构造函数中有两个参数，一个是文件名，另一个是布尔值。如果此值为false，则创建一个新文件，如果存在原文件，则覆盖。如果此值为true，则打开文件保留原来数据，如果找不到文件，则创建新文件。

与创建FileStream对象不同，创建StreamWriter对象不会提供一组类似的选项：除了使用Boolean值添加到文件的末尾或创建新文件之 外，根本没有像FileStream类那样指定FileMode属性的选项。而且，没有设置FileAccess属性的选项，因此总是有对文件的读/写权 限。为了使用高级参数，必须先在FileStream构造函数中指定这些参数，然后在FileStream对象中创建StreamWriter。

using System;

using System.Collections.Generic;

using System.Text;

using System.IO;

class Program

{

static void Main(string[]args)

{

try

{

FileStream aFile = new FileStream(@"c:\123\欢迎.txt", FileMode.OpenOrCreate);

StreamWriter sw = new StreamWriter(aFile);

sw.WriteLine("为今后我们之间的进一步合作，");

sw.WriteLine("为我们之间日益增进的友谊，");

sw.Write("为朋友们的健康幸福，");

sw.Write("干杯！朋友！");

sw.Close();

}

catch (IOException ex)

{

Console.WriteLine(ex.Message);

Console.ReadLine();

return ;

}

}

}

编译并运行该项目。如果没有发现错误，则项目会很快运行，并关闭。因为我们在控制台上没有显示任何内容，所以在控制台中无法看到程序的执行情况。

这个简单的应用程序演示了StreamWriter类的两个最重要的方法：Write()和WriteLine()。这两个方法具有许多重载的版本，可以完成更高级的文件输出，但是本示例只使用基本的字符串输出。

WriteLine()方法会写入传递给它的字符串，其后跟有换行符。

Write()方法只是把传送给它的字符串写入文件，但不追加换行符，因此可以使用多个Write()语句写入完整的句子或段落。