

## Lab12 IO

学习目标：

### 1.掌握文件读写操作

#### 一、文件读写

数据流（也称为流）是一个用于传输数据的对象。数据的传输有两个方向，如果是数据从外部源传输到程序中，就称为读取流；如果是数据从程序传输到外部源，就称为写入流。

在 `System.IO` 命名空间中，对文件的读写操作是依靠流操作进行的。也就是说.NET 中一个被打开以后的文件就是一个数据流，对已经找开的文件进行读写操作就是在对内存中的一个数据流进行读写操作。

##### 1.StreamReader 类

`StreamReader` 类用于从文件中读取数据，该类是一个通用类，可用于任何流。`StreamReader` 类以一种特定的编码输入字符，默认的编码为 `UTF-8`，`UTF-8` 可以正确处理 `Unicode` 字符并在操作系统的本地化版本上提供一致的结果。`StreamReader` 类的常用方法如下表所示。

StreamReader 类的常用方法

方法	说明
<code>Close</code>	关闭 <code>StreamReader</code> 对象和基础流
<code>Dispose</code>	释放所有 <code>StreamReader</code> 对象资源
<code>Peek</code>	返回下一个可用的字符
<code>Read</code>	读取输入流中的下一个字符或下一组字符
<code>ReadLine</code>	从数据流中读取一行数据，并作为字符串返回

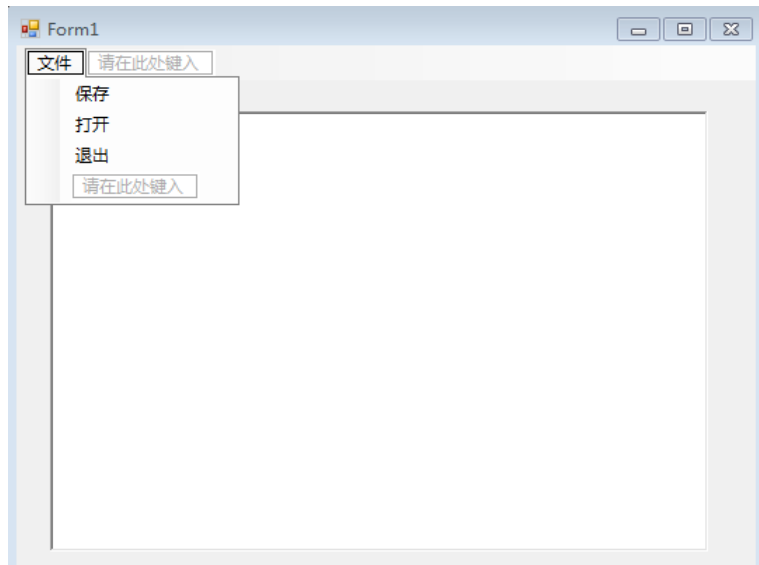
##### 2.StreamWriter 类

`StreamWriter` 类用于将字符和字符串写入文件，它实际上也是先转换成 `FileStream` 对象，然后向文件中写入数据的，所以在创建对象时可以通过 `FileStream` 对象来创建 `StreamWriter` 对象，同时也可以直接创建 `StreamWriter` 对象。`StreamWriter` 默认使用 `UTF8Encoding` 编码。`StreamWriter` 类的常用方法如下表所示。

StreamWriter 类的常用方法

方法	说明
<code>Close</code>	关闭 <code>StreamWriter</code> 对象和基础流
<code>Dispose</code>	释放所有 <code>StreamWriter</code> 对象资源
<code>Flush</code>	清理当前编写器的所有缓冲区，并使所有缓冲数据写入基础流
<code>Write</code>	写入流
<code>WriteLine</code>	写入指定的某些数据，后跟行结束符

项目练习：完成文件读写项目，效果图如下图所示。



### 1 保存代码

```
private void miSave_Click(object sender, EventArgs e)
{
    OpenFileDialog dialog = new OpenFileDialog();
    if (dialog.ShowDialog() == DialogResult.OK)
    {
        string name = dialog.FileName;
        StreamWriter writer = new StreamWriter(name);
        writer.Write(richTextBox1.Text);
        writer.Close();
        MessageBox.Show("保存成功");
    }
}
```

### 2.打开代码

```
private void miOpen_Click(object sender, EventArgs e)
{
    OpenFileDialog dialog = new OpenFileDialog();
    if (dialog.ShowDialog() == DialogResult.OK)
    {
        string name = dialog.FileName;
        StreamReader reader = new StreamReader(name);
        richTextBox1.Text = reader.ReadToEnd();
        reader.Close();
    }
}
```

### 3.退出代码

```
private void miExit_Click(object sender, EventArgs e)
{
    this.Close();
}
```

## 二、酒店管理系统项目练习

为 CheckOutForm 窗体中的“导出”按钮添加事件处理程序。

```
private void btnPrint_Click(object sender, EventArgs e)
{
    //文件路径
    string path = "D:\\发票单\\"+DateTime.Now.ToString
("yyyyMMddHHmmss")+ ".txt";
    try
    {
        StreamWriter sw = new StreamWriter(path);
        sw.WriteLine("-----黑龙江科技大学专家公寓-----");
        sw.WriteLine("客户:"+txtClientName.Text);
        sw.WriteLine("住宿费"+txtTotal.Text);
        sw.WriteLine("开票日期"+DateTime.Now.ToString("yyyy-MM-dd"));
        sw.Close();
        MessageBox.Show("成功导出");
    }
    catch (Exception)
    {
        MessageBox.Show("导出失败");
    }
}
```