## IO

流动的是数据---用于数据传输的API。IO---Input/Output--当数据从外部流向程序---输入流；数据从程序流向外部的时候---输出流

根据传输方向：输入和输出

根据传输形式：字节和字符

|  |  |  |
| --- | --- | --- |
|  | 输入流 | 输出流 |
| 字节流 | InputStream | OutputStream |
| 字符流 | Reader | Writer |

四个基本流对应的类都是抽象类

数据的来源：硬盘、网络、内存、输入设备

向TXT文件中去写入一个字符串---输出流 字符流 和文件有关的流---FileWriter

FileWriter---会创建一个新的文件。在写出数据的时候会写到缓冲区中，当缓冲区满了之后才会放到文件中，所以如果缓冲区没有满而程序有已经结束，一部分数据会死在缓冲区中，因此需要进行冲刷缓冲区的操作。

### 流中异常处理

1. 将流对象外置定义，内置初始化

2. 需要判断流对象是否为null

3. 在关流之后需要将流对象强制置为null

4. 需要在try中写完数据之后进行一步冲刷缓冲区的操作

### FileReader

字符输入流没有缓冲区

一般习惯上会手动指定一个缓冲区

练习：复制文件

注意：Java中原生的字符流不允许直接操作Office组件

### BufferedReader---缓冲流

可以利用readLine方法实现按行读取

练习：统计Java代码的行数

需要获当前目录下的所有的子文件和子目录，如果是子文件，判断是否是一个Java文件，如果是Java文件，需要创建一个流指向这个Java文件然后按行读取统计，如果是一个子目录---递归