Atiti 重定向标准输出到字符串转接口adapter stream流体系 以及 重定向到字符串

原理：：syso 向ByteArrayOutputStream这个流理想write字节。。然后可以使用toByteArray()得到字节，转化为字符串即可使用啦。。

## （一）Stream的分类：

1 Node Stream :基本流，可以从名称中看出他是从哪个地方输入输出的。

1.1 用于文件输入输出流： FileInputStream, FileOutputStream

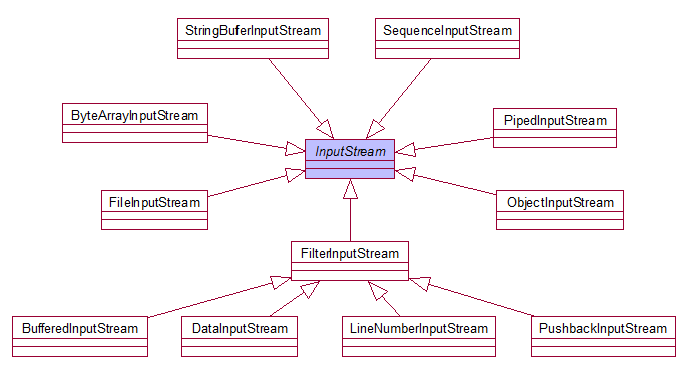
1.2 用于内存数组的输入输出流：ByteArrayInputStream, ByteArrayOutputStream

1.3 用于字符串的输入输出流：StringArrayInputStream, StringArrayOutputStream

1.4 用于管道的输入输出流：PipedInputStream, PipeOutStream (用于线程间的交互)

**二、基于字节的I/O操作(InputStream和OutputStream)**

我们先来看看类图：

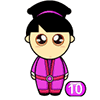


总共就是两个流阿：字节，字符  
   
字节流可用于任何类型的对象，  
而字符流只能处理字符或者字符串，Unicode字符；  
也就是说 字节流可以读写所有的文件，  
而字符流只能读写文本文件。不能读像音频电影之类的  
但是能用字符流的时候就不要用字节流，因为字符流的读写效率更高一些。

 本回答由网友推荐

评论

4 3

[](http://zhidao.baidu.com/usercenter?uid=b0364069236f25705e79a009)

[dngoryaner](http://zhidao.baidu.com/usercenter?uid=b0364069236f25705e79a009" \t "http://zhidao.baidu.com/_blank)

采纳率：31% 来自：[芝麻团](http://zhidao.baidu.com/uteam/?fr=qb" \t "http://zhidao.baidu.com/_blank) 擅长： [生活](http://zhidao.baidu.com/browse/80" \t "http://zhidao.baidu.com/_blank) [电脑/网络](http://zhidao.baidu.com/browse/74" \t "http://zhidao.baidu.com/_blank) [医疗健康](http://zhidao.baidu.com/browse/79" \t "http://zhidao.baidu.com/_blank) [娱乐休闲](http://zhidao.baidu.com/browse/1031" \t "http://zhidao.baidu.com/_blank) [社会民生](http://zhidao.baidu.com/browse/84" \t "http://zhidao.baidu.com/_blank)

## 其他回答

1.因为java要和各种其他的譬如电脑文档等交换信息，就有了输入输出流。  
2.从最简单的输入输出流懂起，以后遇到什么解决什么。  
3.两大类，字节流和字符流  
字节流 ：BufferedInputStream,BufferedOutputStream,FileInputStream ,FileOutputStream  
字符流：BufferedReader,BufferedWriter,FileReader,FileWriter  
4.在读写二进制数据时就会使用字节流。在设计用于处理字符输入输出时用的是Unicode，所以要用字符流，在某些情况下，字符流比字节流更高效。字节流和字符流的功能大部分是并行的。

 StringBufferInputStream in=newStringBufferInputStream(content); 但是这个类已经过时了 并且好像不支持中文

**public** **static** **void** main(String[] args) {

PrintStream stdOut=System.***out***;//保存标准输出流

ByteArrayOutputStream bout=**new** ByteArrayOutputStream();

PrintStream ps = **new** PrintStream(bout);

System.*setOut*(ps);

System.***out***.println("--wanning..for debug out stacktrace");

**byte**[] buf=bout.toByteArray();

String s=**new** String(buf);

//restore std out

System.*setOut*(stdOut);

System.***out***.println("aa"+s);

}

StreamUtil sx = **new** StreamUtil();

sx.RedirectToStrOut();

System.***out***.println("--wanning..for debug out stacktrace");

String s=sx.getStr();

//restore std out

sx.restoreStdOut();

ByteArrayOutputStream用法 - Mayola - 博客园.html