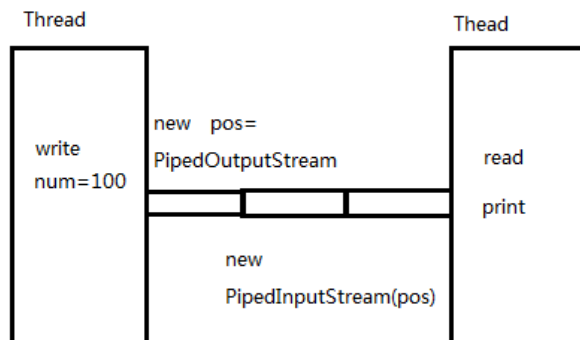


第十九天

1、Filter 相关的流

`DataInputStream/DataOutputStream`: 可以直接对 `java` 中的基本数据类型进行读、写操作
-->`DataStreamTest.java`

`PipedInputStream/PipedOutputStream`: 可以在线程间进行数据的交换-->`PipedStreamTest.java`



`SequenceInputStream`: 可以把多个字节流整合成一个, 顺序读取或写入-->`SequenceTest.java`

`ByteArrayInputStream/ByteArrayOutputStream`: 在内存中创建数组, 当作缓冲区, 最后一次从缓冲区中获取数据-->`ByteArrayTest.java`

2、字符流是可以直接读写字符的 IO 流

字符流读取字符, 就要先读取到字节数据, 然后转为字符; 如果要写出字符, 需要把字符转为字节再写出

`InputStreamReader/OutputStreamWriter`: 按照指定的字符编码进行读写操作
-->`ReaderWriterTest.java`

字节流、字符流转换的桥梁

`BufferedReader` 的 `read()`方法读取字符时会一次读取若干字符到缓冲区, 然后逐个返回给程序, 降低读取文件的次数, 提高效率

`BufferedWriter` 的 `write()`方法写出字符时会先写到缓冲区, 缓冲区写满时才会写到文件, 降低写文件的次数, 提高效率

-->`BufferedReaderWriterTest.java`

`FileReader` 类的 `read()`方法可以按照字符大小读取(使用系统默认字符编码)

`FileWriter` 类(使用系统默认字符编码)的 `write()`方法可以自动把字符转为字节写出(使用系统默认字符编码)

-->`FileReaderWriterTest.java`

练习:

1) 以行为单位将/etc/passwd 内容反转输出 passwd.txt?

-->`ReverseText.java`

2) 统计/etc/passwd 中每个字符出现的次数,将结果写到 count.txt? -->CountCharacter.java

3、File: 在 java 中代表文件系统上的文件或目录-->FileTest.java

public boolean createNewFile():创建文件 如果存在这样的文件,就不创建了
public boolean mkdir():创建文件夹 如果存在这样的目录,就不创建了
public boolean mkdirs():创建文件夹,如果父目录不存在,会帮你创建出来
public boolean renameTo(File dest):把文件重命名为指定的文件路径
public boolean delete():删除文件或者目录,不走回收站,目录下不能有子目录或文件
public boolean isDirectory():判断是否是目录
public boolean isFile():判断是否是文件
public boolean exists():判断是否存在
public boolean canRead():判断是否可读
public boolean canWrite():判断是否可写
public boolean isHidden():判断是否隐藏
public String getAbsolutePath(): 获取绝对路径
public String getPath():获取路径
public String getName():获取名称
public long length():获取长度。字节数
public long lastModified():获取最后一次的修改时间,毫秒值
public String[] list():获取指定目录下的所有文件或者文件夹的名称数组
public File[] listFiles():获取指定目录下的所有文件或者文件夹的 File 数组

练习:

1) 判断目录下是否有.txt 的文件,如果有,就输出该文件名称? -->FileTest.java

2) 删除非空目录? -->DeleteDirectory.java 递归

类似于 `rm -r aa`