2016/12/5 InputStream (2).svg

ByteArrayInputStream

内部用一个buf[], mark, pos, count来实现

mark:做一个位置标记,

便于后续reset()后可以从做得标记位置开始读.

pos:下一次read()的位置.count:有效字符的个数.

FileInputStream

打开一个文件流,

内部用一个FileDescriptor对象来维护与之关联的文件 fd对象里面的handle变量关联着真正的文件.

BufferedInputStream 带缓冲区的InputStream. 内部用buf[], pos, markpos, count实现.

DataInputStream

可以用可移植的方式读取8种基本数据类型. 多线程访问DataInputStream是不安全的. readFully(byte[])会一直读直到byte[]装满. InputStream

FilterInputStream

装饰器模式

PipedInputStream

内部用一个环形缓冲区buffer[]来实现,和PipeOutputStream结合用于管道的读写不建议这两者处在同一个线程里面.

SequenceInputStream

将多个InputStream转化为一个InputStream.

内部用一个 Enumeration<? extends InputStream> e

集合来装所有的InputStream,

读完一个InputStream后调用e.hasMoreElements() 获取下一个InputStream流读.

ObjectInputStream 用于序列化对象用.

CheckedInputStream

从流中读到的字节的CheckSum,用于校验数据完整性. 有CRC32和Adler32算法

DeflaterInputStream

压缩流, 然后把压缩后的流做为新的流, 可以从中读取字节.

GZIPInputStream 解压缩GZIP格式的数据 只能用于单个数据流.

InflaterInputStream

解压缩流, 然后把解压缩后的流作为新的流.

当然,并不是真的解压缩完成后才读,

是一遍解压缩一边读取. 可以从中读取字节.

常常和DeflaterOutputStream一起用.

PushBackInputStream

可以将读到的一部分字节回退到流里面.

内部用一个buf[]实现,回退就是改变buf[]

前面的字节,重新设置pos值

ZipInputStream 解压缩ZIP格式的数据 可以用于多个数据流.