Class java.io.SequenceInputStream

```
java.lang.Object
   |
   +----java.io.InputStream
           |
           +----java.io.SequenceInputStream
```

---

public class **SequenceInputStream**
extends InputStream

Converts a sequence of input streams into an InputStream.

Author:
Author van Hoff
Version:
1.9, 08/11/95

Constructor Index

o **SequenceInputStream**(Enumeration)
Constructs a new SequenceInputStream initialized to the specified list.
o **SequenceInputStream**(InputStream, InputStream)
Constructs a new SequenceInputStream initialized to the two specified input streams.

Method Index

o **close**()
Closes the input stream; flipping to the next stream, if an EOF is reached.
o **read**()
Reads a stream, and upon reaching an EOF, flips to the next stream.
o **read**(byte[], int, int)
Reads data into an array of bytes, and upon reaching an EOF, flips to the next stream.

Constructors

o **SequenceInputStream**

```
public SequenceInputStream(Enumeration e)
```

Constructs a new SequenceInputStream initialized to the specified list.
Parameters:
e – the list

o **SequenceInputStream**

```
public SequenceInputStream(InputStream s1,
                           InputStream s2)
```

Constructs a new SequenceInputStream initialized to the two specified input streams.
Parameters:
s1 – the first input stream
s2 – the second input stream

Methods

o **read**

```
public int read() throws IOException
```

Reads a stream, and upon reaching an EOF, flips to the next stream.
Overrides:
read in class InputStream

o **read**

```
public int read(byte buf[],
                int pos,
                int len) throws IOException
```

Reads data into an array of bytes, and upon reaching an EOF, flips to the next stream.
Parameters:
buf – the buffer into which the data is read
pos – the start position of the data
len – the maximum number of bytes read
Throws: IOException
If an I/O error has occurred.
Overrides:
read in class InputStream

o **close**

```
public void close() throws IOException
```

Closes the input stream; flipping to the next stream, if an EOF is reached. This method must be called to release any resources associated with the stream.

Throws: IOException

If an I/O error has occurred.

Overrides:

close in class InputStream