**Java SequenceInputStream Class**

Java SequenceInputStream class is used to read data from multiple streams. It reads data sequentially (one by one).

Java SequenceInputStream Class declaration

**Let's see the declaration for Java.io.SequenceInputStream class:**

1. **public** **class** SequenceInputStream **extends** InputStream

**Constructors of SequenceInputStream class**

|  |  |
| --- | --- |
| **Constructor** | **Description** |
| SequenceInputStream(InputStream s1, InputStream s2) | creates a new input stream by reading the data of two input stream in order, first s1 and then s2. |
| SequenceInputStream(Enumeration e) | creates a new input stream by reading the data of an enumeration whose type is InputStream. |

**Methods of SequenceInputStream class**

|  |  |
| --- | --- |
| **Method** | **Description** |
| int read() | It is used to read the next byte of data from the input stream. |
| int read(byte[] ary, int off, int len) | It is used to read len bytes of data from the input stream into the array of bytes. |
| int available() | It is used to return the maximum number of byte that can be read from an input stream. |
| void close() | It is used to close the input stream. |

**SAMPLE NFILES**

**Cat.txt : java is my language**

**Cat2.txt : java and c# is the best.**

**EXAMPLE**

|  |
| --- |
| **package** com.company;  **import** java.io.\*;  **public class** Main {   **public static void** main(String[] args) {  **try** {  FileInputStream fileInputStream = **new** FileInputStream(**"D:\\Cat.txt"**);  FileInputStream fileInputStream1 = **new** FileInputStream(**"D:\\Cat2.txt"**);  SequenceInputStream sequenceInputStream =**new** SequenceInputStream(fileInputStream, fileInputStream1);  **int** i = 0;  **while** ((i = sequenceInputStream.read()) != -1){  System.***out***.print((**char**)i);  }  sequenceInputStream.close();  fileInputStream1.close();  fileInputStream.close();  } **catch** (Exception ex){  System.***out***.println(**"Message : "** + ex);  }  } } |
| **java is my languagejava and c# is the best.** |