JAVA.IO.BUFFEREDINPUTSTREAM.CLOSE METHOD

 Copyright © tutorialspoint.com

Description

The **java.io.BufferedInputStream.close** method closes the buffered input stream and releases any system resources associated with the stream. After closing the stream, the **read**, **available**, **skip**, or **reset** invocations will throw I/O Exception.

Invoking close on previously closed stream has no effects.

Declaration

Following is the declaration for **java.io.BufferedInputStream.close** method

```
public void close()
```

Parameters

- **NA**

Return Value

This method does not return any value.

Exception

- **IOException** -- -- if any I/O error occurs.

Example

The following example shows the usage of java.io.BufferedInputStream.close method.

```
package com.tutorialspoint;

import java.io.BufferedInputStream;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.InputStream;

public class BufferedInputStreamDemo {
   public static void main(String[] args) throws Exception {

      InputStream inStream = null;
      BufferedInputStream bis = null;

      try{
         // open input stream test.txt for reading purpose.
         inStream = new FileInputStream("c:/test.txt");

         // input stream is converted to buffered input stream
         bis = new BufferedInputStream(inStream);

         // invoke available
         int byteNum = bis.available();

         // number of bytes available is printed
         System.out.println(byteNum);

         // releases any system resources associated with the stream
         bis.close();

         // throws io exception on available() invocation
         byteNum = bis.available();
```

```
         System.out.println(byteNum);

      } catch (IOException e) {

         // exception occurred.
         System.out.println("Error: Sorry 'bis' is closed");
      }finally{

         // releases any system resources associated with the stream
         if(inStream!=null)
            inStream.close();
      }
   }
}
```

Assuming we have a text file **c:/test.txt**, which has the following content. This file will be used as an input for our example program:

```
ABCDE
```

Let us compile and run the above program, this will produce the following result:

```
5
Error: Sorry 'bis' is closed
```

Loading [MathJax]/jax/output/HTML-CSS/jax.js