JAVA.IO.FILTEROUTPUTSTREAM.WRITE METHOD

 Copyright © tutorialspoint.com

Description

The **java.io.FilterOutputStream.write*byte[]b*** method writes b.length bytes to this output stream.

Declaration

Following is the declaration for **java.io.FilterOutputStream.write*byte[]b*** method:

```
public void write(byte[] b)
```

Parameters

- **b** -- source buffer to be written to the stream

Return Value

This method does not return any value.

Exception

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

Example

The following example shows the usage of java.io.FilterOutputStream.write*byte[]b* method.

```
package com.tutorialspoint;

import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.FilterOutputStream;
import java.io.IOException;
import java.io.OutputStream;

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

      OutputStream os = null;
      FilterOutputStream fos = null;
      FileInputStream fis = null;
      byte[] buffer = {65, 66, 67, 68, 69};
      int i=0;
      char c;

      try{
         // create output streams
         os = new FileOutputStream("C://test.txt");
         fos = new FilterOutputStream(os);

         // writes buffer to the output stream
         fos.write(buffer);

         // forces byte contents to written out to the stream
         fos.flush();

         // create input streams
         fis = new FileInputStream("C://test.txt");

         while((i=fis.read())!=-1)
         {
            // converts integer to the character
```

```
            c = (char)i;

            // prints
            System.out.println("Character read: "+c);
         }
      }catch(IOException e){

         // if any I/O error occurs
         System.out.print("Close() is invoked prior to write()");
      }finally{

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

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

```
Character read: A
Character read: B
Character read: C
Character read: D
Character read: E
```

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