Java.io.BufferedOutputStream class in Java

Difficulty Level : Basic Last Updated : 24 Jan, 2017

Java.io.BufferedInputStream class in Java

Java.io.BufferedOutputStream class implements a buffered output stream. By setting up such an output stream, an application can write bytes to the underlying output stream without necessarily causing a call to the underlying system for each byte written.

Fields

- **protected byte[] buf**: The internal buffer where data is stored.
- **protected int count:** The number of valid bytes in the buffer.

Constructor and Description

- **BufferedOutputStream(OutputStream out) :** Creates a new buffered output stream to write data to the specified underlying output stream.
- **BufferedOutputStream(OutputStream out, int size) :** Creates a new buffered output stream to write data to the specified underlying output stream with the specified buffer size.

Methods:

- **void flush() :** Flushes this buffered output stream.

```
Syntax :public void flush()
           throws IOException
Overrides:
flush in class FilterOutputStream
Throws:
IOException
```

- **void write(byte[] b, int off, int len) :** Writes len bytes from the specified byte array starting at offset off to this buffered output stream.

```
Syntax :
Parameters:
b - the data.
off - the start offset in the data.
len - the number of bytes to write.
Throws:
IOException
```

- **void write(int b) :** Writes the specified byte to this buffered output stream.

```
Syntax :
Parameters:
b - the byte to be written.
Throws:
IOException
```

Program:

```java
//Java program demonstrating BufferedOutputStream

import java.io.*;

class BufferedOutputStreamDemo
{
    public static void main(String args[])throws Exception
    {
        FileOutputStream fout = new FileOutputStream("f1.txt");

        //creating bufferdOutputStream obj
        BufferedOutputStream bout = new BufferedOutputStream(fout);

        //illustrating write() method
        for(int i = 65; i < 75; i++)
        {
            bout.write(i);
        }

        byte b[] = { 75, 76, 77, 78, 79, 80 };
        bout.write(b);

        //illustrating flush() method
        bout.flush();

        //illustrating close() method
        bout.close();
        fout.close();
    }
}
```

Output :

```
ABCDEFGHIJKLMNOP
```

This article is contributed by Nishant Sharma. If you like GeeksforGeeks and would like to contribute, you can also write an article using contribute.geeksforgeeks.org or mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.