

JAVA - BYTEARRAYINPUTSTREAM

http://www.tutorialspoint.com/java/java_bytearrayinputstream.htm

Copyright © tutorialspoint.com

The `ByteArrayInputStream` class allows a buffer in the memory to be used as an `InputStream`. The input source is a byte array. There are following forms of constructors to create `ByteArrayInputStream` objects

Takes a byte array as the parameter:

```
ByteArrayInputStream bArray = new ByteArrayInputStream(byte [] a);
```

Another form takes an array of bytes, and two ints, where **off** is the first byte to be read and **len** is the number of bytes to be read.

```
ByteArrayInputStream bArray = new ByteArrayInputStream(byte []a,  
                                                    int off,  
                                                    int len)
```

Once you have `ByteArrayInputStream` object in hand then there is a list of helper methods which can be used to read the stream or to do other operations on the stream.

SN	Methods with Description
1	public int read() This method reads the next byte of data from the <code>InputStream</code> . Returns an int as the next byte of data. If it is end of file then it returns -1.
2	public int read(byte[] r, int off, int len) This method reads upto len number of bytes starting from off from the input stream into an array. Returns the total number of bytes read. If end of file -1 will be returned.
3	public int available() Gives the number of bytes that can be read from this file input stream. Returns an int that gives the number of bytes to be read.
4	public void mark(int read) This sets the current marked position in the stream. The parameter gives the maximum limit of bytes that can be read before the marked position becomes invalid.
5	public long skip(long n) Skips n number of bytes from the stream. This returns the actual number of bytes skipped.

Example:

Following is the example to demonstrate `ByteArrayInputStream` and `ByteArrayOutputStream`

```
import java.io.*;  
  
public class ByteStreamTest {  
  
    public static void main(String args[]) throws IOException {
```

```

ByteArrayOutputStream bOutput = new ByteArrayOutputStream(12);

while( bOutput.size() != 10 ) {
    // Gets the inputs from the user
    bOutput.write(System.in.read());
}

byte b [] = bOutput.toByteArray();
System.out.println("Print the content");
for(int x= 0 ; x < b.length; x++) {
    // printing the characters
    System.out.print((char)b[x] + " ");
}
System.out.println(" ");

int c;

ByteArrayInputStream bInput = new ByteArrayInputStream(b);

System.out.println("Converting characters to Upper case ");
for(int y = 0 ; y < 1; y++ ) {
    while(( c= bInput.read()) != -1) {
        System.out.println(Character.toUpperCase((char)c));
    }
    bInput.reset();
}
}
}

```

Here is the sample run of the above program:

```

asdfghjkl y
Print the content
a s d f g h j k l y
Converting characters to Upper case
A
S
D
F
G
H
J
K
L
Y

```