

BufferedReader

java.io

```
java.lang.Object
  java.io.Reader
    java.io.BufferedReader
```

Kılğıladığı Arayüzler:

Closeable, AutoCloseable, Readable

Altsınıfları:

LineNumberReader

Bildirimi:

```
public class BufferedReader
    extends Reader
```

Giriş akımından okuduğu text'ten karakterleri, array'leri ya da satırları buffer'a yazar. Etkili bir giriş aracıdır. Buffer uzunluğu öntanımlı olarak kullanılabileceği gibi, istenirse uzunluk belirtilebilir. Çoğu uygulamada öntanımlı uzunluk yeterlidir. Öntanımlı uzunluk JDK sürümlerinde farklı olabilir; ama çoğu sürümde 8192 byte olarak tanımlıdır.

Genel kural olarak, BufferedReader tarafından okunan veri, ilgili karakter ya da byte dizisine dönüşür. Kaynaktaki veri büyükse, her okuma sonunda söz konusu dönüşümün yapılması, eylemi yavaşlatacaktır. O nedenle, doğrudan read() metodu ya da readLine() metodu yerine FileReader ya da InputStreamReader sınıflarını eyleme katmak daha etkin olur. Örneğin,

```
BufferedReader in
    = new BufferedReader(new FileReader("dosya.in"));
```

kodları, dosyayı bir bütün olarak buffer'a koyar. Oysa doğrudan read() ya da ReadLine() metotları kullanılırsa, her okuma sonunda buffer'a giren verilerin byte ya da karaktere dönüşmesi gerekir.

DataInputStream sınıfını kullanan programlarda, onun yerine uygun bir BufferedReader konularak, text girişleri yerelleştirilebilir (localize).

Veri Alanları:

BufferedReader sınıfının, yalnızca java.io.Reader sınıfından kalıtsal gelen

lock

değişkeni vardır.

Kurucuları:

BufferedReader (Reader in)

Creates a buffering character-input stream that uses a default-sized input buffer.

BufferedReader (Reader in, int sz)

Creates a buffering character-input stream that uses an input buffer of the specified size.

BufferedReader Sınıfının Metotları:

void **close**()

Closes the stream and releases any system resources associated with it.

void **mark**(int readAheadLimit)

Marks the present position in the stream.

boolean **markSupported**()

Tells whether this stream supports the mark() operation, which it does.

int **read**()

Reads a single character.

int **read**(char[] cbuf, int off, int len)

Reads characters into a portion of an array.

String **readLine**()

Reads a line of text.

boolean **ready**()

Tells whether this stream is ready to be read.

void **reset**()

Resets the stream to the most recent mark.

long **skip**(long n)

Skips characters.

java.io.Reader Sınıfından Kalıtsal Gelen Metotlar:

read, read

java.lang.Object sınıfından kalıtsal gelen metotlar:

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

InputStream tipinden olan `System.in` (klavye)'den bir satır okumak için onu `BufferedReader` içine gömeriz. Bunu yapmak için, öncelikle onu `InputStreamReader` içine gömüyoruz.

```
package BufferedReader;

import java.io.*;

class Demo {
    public static void main(String[] args) throws IOException {
        InputStreamReader isr = new InputStreamReader(System.in);
        BufferedReader input = new BufferedReader(isr);

        String line; // klavyeden girilen satırları tutar

        System.out.println("Lütfen bir satır giriniz :");
        while ((line = input.readLine()) != null) {
            System.out.println(line);
        }
    }
}
```

```
/**
 Lütfen bir satır giriniz :
 a1b2c3 d4 d5 f6 g7 h 8 ı 9 i 10 j11
 a1b2c3 d4 d5 f6 g7 h 8 ı 9 i 10 j11
 */
```

Aşağıdaki programda, `InputStream` tipinden olan `System.in` standart girişinden (klavye) bir seferde bir satır okumak için, önce `InputStreamReader` sınıfına gömülüyor. Sonra o da `BufferedReader` sınıfına gömülüyor. `main()` metoduna hata yakalama deyiminin eklendiğine dikkat ediniz.

```
package BufferedReader;

import java.io.*;

class Demo {
```

```

public static void main(String[] args) throws IOException {

    InputStreamReader isr = new InputStreamReader(System.in);
    BufferedReader br = new BufferedReader(isr);

    String satır; // satır tutan değişken
    System.out.println("Lütfen bir satır giriniz:");

    while ((satır = br.readLine()) != null) {
        System.out.println(satır);
    }
}
/**
 */

```

Aşağıdaki program, klavyeden girilen satırı sayıya dönüştürüyor.

```

import java.io.*;

public class Demo {
    public static void main(String[] args) throws IOException {
        BufferedReader stdin = new BufferedReader(new InputStreamReader(
            System.in));

        double root1, root2;
        double rootca, rootcb;

        System.out.println(" ax^2 + bx + c = denkleminin köklerini bulur.");
        System.out.println(" a katsayısını giriniz :");
        int a = Integer.parseInt(stdin.readLine());

        System.out.println("b katsayısını giriniz :");
        int b = Integer.parseInt(stdin.readLine());

        System.out.println("c sabitini giriniz :");
        int c = Integer.parseInt(stdin.readLine());

        double delta, deltasqrt;

        delta = b * b - 4 * a * c;

        if (delta >= 0) {
            deltasqrt = Math.sqrt(delta);
            root1 = (-b + deltasqrt) / (2 * a);
            root2 = (-b - deltasqrt) / (2 * a);
            System.out.println("Kökler : " + root1 + " and " + root2);
        }

        else {
            deltasqrt = Math.sqrt(-delta);
            rootca = deltasqrt / (2 * a);
            rootcb = deltasqrt / (2 * a);
            System.out.println("Kökler : " + -b / (2 * a) + "+"
                + rootca + " ve " + -b / (2 * a) + "-i" + rootcb);
        }
    }
}

```

```

    }
}

/**
ax^2 + bx + c = denkleminin köklerini bulur.
a katsayısını giriniz :
2
b katsayısını giriniz :
-3
c sabitini giriniz :
4
Kökler : 0+i1.1989578808281798 ve 0-i1.1989578808281798
*/

```

Program: Klavyeden Buffer'a Giriş

Aşağıdaki program `BufferedReader` sınıfının `read()` metodu ile klavyeden girilen metni karakter karakter okuyor ve standart çıkışa gönderiyor. `System.in` akımı `InputStreamReader` sınıfına, o da `BufferedReader` sınıfına gömülmüştür.

```

import java.io.*;

class Demo {
    public static void main(String args[]) throws IOException {
        char c;
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        System.out.println("Bir metin satır giriniz.");

        // karakter oku
        do {
            c = (char) br.read();
            System.out.print(c);
        } while (c != -1);
    }
}

```

```

/**
Bir satır giriniz.
abc 123 def 456 ghi 89.3 nmö
abc 123 def 456 ghi 89.3 nmö
*/

```

Aşağıdaki program klavyeden girilen satırları okuyup ekrana yazıyor.

```

import java.io.*;

class Demo {
    public static void main(String args[]) throws IOException {
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        String str;
    }
}

```

```

        System.out.println("Birkaç satır griniz.");
        System.out.println("Durmak için son satıra 'dur' yazınız.");
        do {
            str = br.readLine();
            System.out.println(str);
        } while (!str.equals("dur"));
    }
}

```

```

/*
Birkaç satır griniz.
Durmak için son satıra 'dur' yazınız.
hjsb
hjsb
oijeoi
oijeoi
6tr7
6tr7
dur
dur
*/

```

Program: Klavyeden BufferedReader'a Giriş

```

// Basit bir editor.
import java.io.*;

class TinyEdit {
    public static void main(String args[])
        throws IOException
    {
        // create a BufferedReader using System.in
        BufferedReader br = new BufferedReader(new
            InputStreamReader(System.in));
        String str[] = new String[100];

        System.out.println("Bir satır yaz.");
        System.out.println("Durmak için yeni bir satıra 'dur' yaz.");
        for(int i=0; i<100; i++) {
            str[i] = br.readLine();
            if(str[i].equals("dur")) break;
        }

        System.out.println("\n Yazdığınız metin budur:");

        // display the lines
        for(int i=0; i<100; i++) {
            if(str[i].equals("dur")) break;
            System.out.println(str[i]);
        }
    }
}

```

```
}  
}
```

```
import java.io.*;  
  
public class Demo {  
    public static void main(String args[]) {  
        int ch = 0;  
        int[] numbers = { 17, 5, 14, 23, 70 };  
        try {  
  
            System.out.println("1. Veri yaz");  
            System.out.println("2. Veri Oku");  
            System.out.println("Lütfen seçiminizi yapınız ");  
            BufferedReader br = new BufferedReader(new InputStreamReader(  
                System.in));  
            ch = Integer.parseInt(br.readLine());  
            switch (ch) {  
                case 1:  
                    FileOutputStream fos = new FileOutputStream("datafile.txt");  
                    BufferedOutputStream bos = new BufferedOutputStream(fos);  
                    DataOutputStream out = new DataOutputStream(bos);  
                    for (int i = 0; i < numbers.length; i++) {  
                        out.writeInt(numbers[i]);  
                    }  
                    System.out.print("write successfully");  
                    out.close();  
                case 2:  
                    FileInputStream fis = new FileInputStream("datafile.txt");  
                    BufferedInputStream bis = new BufferedInputStream(fis);  
                    DataInputStream in = new DataInputStream(bis);  
                    while (true) {  
                        System.out.print(in.readInt());  
                    }  
                default:  
                    System.out.println("Invalid choice");  
            }  
        } catch (Exception e) {  
            System.err.println("Error in read/write data to a file: " + e);  
        }  
    }  
}
```

```
// Text okuma  
import java.io.BufferedReader;  
import java.io.FileReader;  
import java.io.IOException;  
import java.io.File;
```

```

public class TextReader {
    private static void readFile(String fileName) {
        try {
            File file = new File(fileName);
            FileReader reader = new FileReader(file);
            BufferedReader in = new BufferedReader(reader);
            String string;
            while ((string = in.readLine()) != null) {
                System.out.println(string);
            }
            in.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    public static void main(String[] args) {
        if (args.length != 1) {
            System.err.println("Çalıştırma: java TextReader " + "dosya adresi");
            System.exit(0);
        }
        readFile(args[0]);
    }
}

```

```

import java.io.*;

class Demo {
    public static void main(String[] args) throws IOException {

        InputStreamReader isr = new InputStreamReader(System.in);
        BufferedReader br = new BufferedReader(isr);

        String line; // satır tutan değişken

        while ((line = br.readLine()) != null) {
            System.out.println(line);
        }
    }
}

```

```

// BufferedReader örneği
import java.io.*;

class ReadKeys4 {
    public static void main(String args[]) {
        Reader in = new BufferedReader(new InputStreamReader(System.in));
        char buf[] = new char[10];
        try {

```



```
        in.read(buf, 0, 10);
    } catch (Exception e) {
        System.out.println("Error: " + e.toString());
    }
    String s = new String(buf);
    System.out.println(s);
}
}
```