java.io

**Class DataInputStream**

* [java.lang.Object](https://docs.oracle.com/javase/8/docs/api/java/lang/Object.html)
  + [java.io.InputStream](https://docs.oracle.com/javase/8/docs/api/java/io/InputStream.html)
    - [java.io.FilterInputStream](https://docs.oracle.com/javase/8/docs/api/java/io/FilterInputStream.html)
      * java.io.DataInputStream
* **All Implemented Interfaces:**

[Closeable](https://docs.oracle.com/javase/8/docs/api/java/io/Closeable.html), [DataInput](https://docs.oracle.com/javase/8/docs/api/java/io/DataInput.html" \o "interface in java.io), [AutoCloseable](https://docs.oracle.com/javase/8/docs/api/java/lang/AutoCloseable.html" \o "interface in java.lang)

本篇所有的api英文解析,请参照DataInput接口中的api文档

DataInputStream不是现成安全的,他的一个主要特点是可以读取基本**数据类型**.而且他的类中没有读取**基本数据类型**的方法.

### *Constructor Summary*

|  |
| --- |
| **构造方法** |
| **Constructor and Description** |
| [**DataInputStream**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#DataInputStream-java.io.InputStream-)([**InputStream**](https://docs.oracle.com/javase/8/docs/api/java/io/InputStream.html) in) |

|  |  |
| --- | --- |
| int | [**read**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#read-byte:A-)(byte[] b)  从流中读取一个字节,放到b中(b中原先的内容会被覆盖),并返回放入到b中的字节的个数  b is null抛空指针。如果b1当前的length为0，返回0.如果从流中读不到内容了，返回-1 |

|  |
| --- |
| 实例 |
| **public** **static** **void** main(String[] args) **throws** IOException {  **byte** [] b = {95,96,97};  **byte** [] b1 =**new** **byte**[5];  ByteArrayInputStream in = **new** ByteArrayInputStream(b);  DataInputStream dis = **new** DataInputStream(in);  **int** i = dis.read(b1); //把b中的95,96,97放到b1中。  System.***out***.println(i); //3  } |

|  |  |
| --- | --- |
| int | [**read**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#read-byte:A-int-int-)(byte[] b, int off, int len)  从流中读取len个字节,然后放到b中.从b中下标为off的位置开始往里放 |

|  |
| --- |
| 实例 |
| **public** **static** **void** main(String[] args) **throws** IOException {  **byte** [] b = {95,96,97};  **byte** [] b2 =**new** **byte**[5];  ByteArrayInputStream in = **new** ByteArrayInputStream(b);  DataInputStream dis = **new** DataInputStream(in);  **int** i = dis.read(b2,1,2); // b1中的内容为[0, 95, 96, 0, 0]  System.***out***.println(i); // 2  } |

|  |  |
| --- | --- |
| boolean | [**readBoolean**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readBoolean--)()  从流中读一个字节，如果不是0返回true，是0返回false。其他抛异常. |

|  |
| --- |
| 实例 |
| **public** **static** **void** main(String[] args) **throws** IOException {  **byte** [] b = {0,96,97};  ByteArrayInputStream in = **new** ByteArrayInputStream(b);  DataInputStream dis = **new** DataInputStream(in);  **boolean** noneZero = dis.readBoolean();// 因为b中第一个字节是0，所以返回false  System.***out***.println(noneZero); //false  } |

|  |  |
| --- | --- |
| byte | [**readByte**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readByte--)()  从流中读取一个字节（signed 8-bit byte.）读取的这个值的范围是-128到127 |

|  |
| --- |
| 实例 |
| **public** **static** **void** main(String[] args) **throws** IOException {  **byte** [] b = {96,97,98};  ByteArrayInputStream in = **new** ByteArrayInputStream(b);    DataInputStream dis = **new** DataInputStream(in);  **byte** byteValue = dis.readByte(); //读取b中一个字节  System.***out***.println(byteValue); //96    byteValue = dis.readByte();  System.***out***.println(byteValue); //97  } |

|  |  |
| --- | --- |
| char | [**readChar**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readChar--)()  从流中读取一个字符.内部就是用了两次read方法,然后把结果转成char |

|  |
| --- |
| 实例 |
| **public** **static** **void** main(String[] args) **throws** IOException {  FileOutputStream fos = **new** FileOutputStream(**new** File("C:\\Users\\liuhb\\Desktop\\test.txt"));  FileInputStream fis = **new** FileInputStream("C:\\Users\\liuhb\\Desktop\\test1.txt");  DataInputStream dis = **new** DataInputStream(fis);  **while** (0 != dis.available()) {  fos.write(dis.readChar());  }  } |

将test1中的【你好】两个字，保存到test中

|  |  |
| --- | --- |
| double | [**readDouble**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readDouble--)()  从流中读取8个字节，然后转成double类型返回 |

|  |
| --- |
| 实例 |
| **public** **static** **void** main(String[] args) **throws** IOException {  **byte** [] b = {96,97,98,99,96,97,98,99};  ByteArrayInputStream in = **new** ByteArrayInputStream(b);  DataInputStream dis = **new** DataInputStream(in);  Double value = dis.readDouble(); //读取b中8个字节  System.***out***.println(value); //1.8646859392855472E156  } |

|  |  |
| --- | --- |
| float | [**readFloat**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readFloat--)()  从流中读取4个字节，然后先转成int类型，然后在强转成float类型返回。 |

|  |
| --- |
| 实例 |
| **public** **static** **void** main(String[] args) **throws** IOException {  **byte** [] b = {96,97,98,99,96,97,98,99};  ByteArrayInputStream in = **new** ByteArrayInputStream(b);  DataInputStream dis = **new** DataInputStream(in);  **float** value = dis.readFloat(); //读取b中4个字节先转成int，在强转为float  System.***out***.println(value); //6.496261E19  } |

|  |  |
| --- | --- |
| void | [**readFully**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readFully-byte:A-)(byte[] b)  b的容量有多大，就从流中获取相应容量的字节，然后填满b  如果还没填满b，流就已经到最后了，那么就报EOF异常 |

|  |
| --- |
| 实例 |
| **public** **static** **void** main(String[] args) **throws** IOException {  **byte** [] b = {96,97,98,99,96,97,98,99};  **byte** [] b1 =**new** **byte**[5];  ByteArrayInputStream in = **new** ByteArrayInputStream(b);  DataInputStream dis = **new** DataInputStream(in);  dis.readFully(b1); //[96, 97, 98, 99, 96]  System.***out***.println(b1.length); // 5  } |

|  |  |
| --- | --- |
| void | [**readFully**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readFully-byte:A-int-int-)(byte[] b, int off, int len)  从流中读取len个字节，然后从下标为off的位置开始存储读取的字节 |

|  |
| --- |
| 实例 |
| **public** **static** **void** main(String[] args) **throws** IOException {  **byte** [] b = {96,97,98,99,96,97,98,99};  **byte** [] b1 =**new** **byte**[5];  ByteArrayInputStream in = **new** ByteArrayInputStream(b);  DataInputStream dis = **new** DataInputStream(in);  dis.readFully(b1,1,3); // [0, 96, 97, 98, 0]  } |

|  |  |
| --- | --- |
| int | [**readInt**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readInt--)()  从流中读取4个字节，并转化为int型返回 |
| [**String**](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html) | [**readLine**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readLine--)()  不建议使用。使用如下方式代替  BufferedReader d  = new BufferedReader(new InputStreamReader(in)); |
| long | [**readLong**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readLong--)()  从流中读取8个字节，并转化为long型返回 |
| short | [**readShort**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readShort--)()  从流中读取2个字节，并转化为short型返回 |

|  |  |
| --- | --- |
| int | [**readUnsignedByte**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readUnsignedByte--)()  从流中读取正数，如果是负数抛异常  所谓unsignedbyte就是指将原先byte类型可以表示的范围-128-127中的负数部分(-128到-1)用正数表示.例如  原先的负数,如-1,-128就不能再用再这样表示了,就需要转化为对应的正数来表示,也就是127之后的值来表示负数.例如-1就要用255表示了.-1就用128来表示 |
|  |  |
| int | [**readUnsignedShort**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readUnsignedShort--)()  同上 |

|  |
| --- |
| 实例 |
| **public** **static** **void** main(String[] args) **throws** IOException {  **byte** [] b = {-128,-1,98,99,96,97,98,99};  ByteArrayInputStream in = **new** ByteArrayInputStream(b);  DataInputStream dis = **new** DataInputStream(in);  **int** value = dis.readUnsignedByte();  System.***out***.println(value); // 128  value = dis.readUnsignedByte();  System.***out***.println(value); // 255  } |

|  |  |
| --- | --- |
| [**String**](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html) | [**readUTF**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readUTF--)()  See the general contract of the readUTF method of DataInput. |
| static [**String**](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html) | [**readUTF**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#readUTF-java.io.DataInput-)([**DataInput**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInput.html) in)  Reads from the stream in a representation of a Unicode character string encoded in [**modified UTF-8**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInput.html#modified-utf-8) format; this string of characters is then returned as a String. |
| int | [**skipBytes**](https://docs.oracle.com/javase/8/docs/api/java/io/DataInputStream.html#skipBytes-int-)(int n)  从流中跳过n个字节,返回实际跳过的自己数 |

|  |
| --- |
| 实例 |
| **public** **static** **void** main(String[] args) **throws** IOException {  **byte** [] b = {-128,-1,98,99,96,97,98,99};  ByteArrayInputStream in = **new** ByteArrayInputStream(b);  DataInputStream dis = **new** DataInputStream(in);  **int** value = dis.skipBytes(2);  System.***out***.println(value); // 2  value = dis.read();  System.***out***.println(value); // 98  } |