

Prog1Tools

Class IOTools

java.lang.Object
 extended by **Prog1Tools.IOTools**

```
public class IOTools
extends java.lang.Object
```

Diese Klasse stellt einige einfache Methoden zum Einlesen von der Tastatur zur Verfügung. Es werden diverse Werte von der Tastatur eingelesen, die jeweils durch ein Leerzeichen, einen Tabstop oder ein Zeilenendezeichen getrennt sein müssen.

Method Summary	
static void	flush() Löscht alles, was sich in der momentanen Zeile befindet.
static boolean	readBoolean() Liest einen boolschen Wert von der Tastatur ein.
static boolean	readBoolean (java.lang.String prompt) Liest einen boolschen Wert von der Tastatur ein.
static char	readChar() Liest ein Zeichen von der Tastatur ein.
static char	readChar (java.lang.String prompt) Liest ein Zeichen von der Tastatur ein.
static double	readDouble() Liest eine double-Zahl von der Tastatur ein.
static double	readDouble (java.lang.String prompt) Liest eine double-Zahl von der Tastatur ein.
static float	readFloat() Liest eine float-Zahl von der Tastatur ein.
static float	readFloat (java.lang.String prompt) Liest eine float-Zahl von der Tastatur ein.
static int	readInt() Liest eine int-Zahl von der Tastatur ein.
static int	readInt (java.lang.String prompt) Liest eine int-Zahl von der Tastatur ein.
static int	readInteger() Liest eine int-Zahl von der Tastatur ein.
static int	readInteger (java.lang.String prompt) Liest eine int-Zahl von der Tastatur ein.
static java.lang.String	readLine() Liest eine ganze Textzeile von der Tastatur ein.

static java.lang.String	readLine (java.lang.String prompt) Liest eine ganze Textzeile von der Tastatur ein.
static long	readLong () Liest eine long-Zahl von der Tastatur ein.
static long	readLong (java.lang.String prompt) Liest eine long-Zahl von der Tastatur ein.
static short	readShort () Liest eine short-Zahl von der Tastatur ein.
static short	readShort (java.lang.String prompt) Liest eine short-Zahl von der Tastatur ein.
static java.lang.String	readString () Liest ein Textwort von der Tastatur ein.
static java.lang.String	readString (java.lang.String prompt) Liest ein Textwort von der Tastatur ein.
static java.lang.String	toString (double d) Wandelt eine double-Zahl in einen String um.

Methods inherited from class java.lang.Object

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

Method Detail

flush

```
public static void flush()
```

Löscht alles, was sich in der momentanen Zeile befindet. Das heißt es wird der Eingabe bis zum Zeilenende keine Beachtung mehr geschenkt

readLine

```
public static java.lang.String readLine(java.lang.String prompt)
```

Liest eine ganze Textzeile von der Tastatur ein. Soll vorher eine Eingabeaufforderung gemacht werden, geschieht dies durch den Parameter. Dieser kann jedoch auch wegfallen.

Parameters:

prompt - eine eventuelle Eingabeaufforderung

Returns:

die eingelesene Zeile.

readInteger

```
public static int readInteger(java.lang.String prompt)
```

Liest eine int-Zahl von der Tastatur ein. Soll vorher eine Eingabeaufforderung gemacht werden, geschieht dies durch den Parameter. Dieser kann jedoch auch wegfallen.

Parameters:

`prompt` - eine eventuelle Eingabeaufforderung

Returns:

die eingelesene Zahl.

readLong

```
public static long readLong(java.lang.String prompt)
```

Liest eine `long`-Zahl von der Tastatur ein. Soll vorher eine Eingabeaufforderung gemacht werden, geschieht dies durch den Parameter. Dieser kann jedoch auch wegfallen.

Parameters:

`prompt` - eine eventuelle Eingabeaufforderung

Returns:

die eingelesene Zahl.

readDouble

```
public static double readDouble(java.lang.String prompt)
```

Liest eine `double`-Zahl von der Tastatur ein. Soll vorher eine Eingabeaufforderung gemacht werden, geschieht dies durch den Parameter. Dieser kann jedoch auch wegfallen.

Parameters:

`prompt` - eine eventuelle Eingabeaufforderung

Returns:

die eingelesene Zahl.

readFloat

```
public static float readFloat(java.lang.String prompt)
```

Liest eine `float`-Zahl von der Tastatur ein. Soll vorher eine Eingabeaufforderung gemacht werden, geschieht dies durch den Parameter. Dieser kann jedoch auch wegfallen.

Parameters:

`prompt` - eine eventuelle Eingabeaufforderung

Returns:

die eingelesene Zahl.

readShort

```
public static short readShort(java.lang.String prompt)
```

Liest eine `short`-Zahl von der Tastatur ein. Soll vorher eine Eingabeaufforderung gemacht werden, geschieht dies durch den Parameter. Dieser kann jedoch auch wegfallen.

Parameters:

`prompt` - eine eventuelle Eingabeaufforderung

Returns:

die eingelesene Zahl.

readBoolean

```
public static boolean readBoolean(java.lang.String prompt)
```

Liest einen boolschen Wert von der Tastatur ein. Soll vorher eine Eingabeaufforderung gemacht werden, geschieht dies durch den Parameter. Dieser kann jedoch auch wegfallen.

Parameters:

prompt - eine eventuelle Eingabeaufforderung

Returns:

der eingelesene Wert.

readString

```
public static java.lang.String readString(java.lang.String prompt)
```

Liest ein Textwort von der Tastatur ein. Soll vorher eine Eingabeaufforderung gemacht werden, geschieht dies durch den Parameter. Dieser kann jedoch auch wegfallen.

Parameters:

prompt - eine eventuelle Eingabeaufforderung

Returns:

das eingelesene Wort.

readChar

```
public static char readChar(java.lang.String prompt)
```

Liest ein Zeichen von der Tastatur ein. Soll vorher eine Eingabeaufforderung gemacht werden, geschieht dies durch den Parameter. Dieser kann jedoch auch wegfallen.

Parameters:

prompt - eine eventuelle Eingabeaufforderung

Returns:

das eingelesene Zeichen.

readLine

```
public static java.lang.String readLine()
```

Liest eine ganze Textzeile von der Tastatur ein.

Returns:

die eingelesene Zeile.

readInteger

```
public static int readInteger()
```

Liest eine int-Zahl von der Tastatur ein.

Returns:

die eingelesene Zahl.

readInt

```
public static int readInt()
```

Liest eine `int`-Zahl von der Tastatur ein.

Returns:

die eingelesene Zahl.

readInt

```
public static int readInt(java.lang.String prompt)
```

Liest eine `int`-Zahl von der Tastatur ein. Soll vorher eine Eingabeaufforderung gemacht werden, geschieht dies durch den Parameter. Dieser kann jedoch auch wegfallen.

Parameters:

`prompt` - eine eventuelle Eingabeaufforderung

Returns:

die eingelesene Zahl.

readLong

```
public static long readLong()
```

Liest eine `long`-Zahl von der Tastatur ein.

Returns:

die eingelesene Zahl.

readDouble

```
public static double readDouble()
```

Liest eine `double`-Zahl von der Tastatur ein.

Returns:

die eingelesene Zahl.

readShort

```
public static short readShort()
```

Liest eine `short`-Zahl von der Tastatur ein.

Returns:

die eingelesene Zahl.

readFloat

```
public static float readFloat()
```

Liest eine `float`-Zahl von der Tastatur ein.

Returns:

die eingelesene Zahl.

readChar

```
public static char readChar()
```

Liest ein Zeichen von der Tastatur ein.

Returns:

das eingelesene Zeichen

readString

```
public static java.lang.String readString()
```

Liest ein Textwort von der Tastatur ein.

Returns:

das eingelesene Wort.

readBoolean

```
public static boolean readBoolean()
```

Liest einen boolschen Wert von der Tastatur ein.

Returns:

das eingelesene Wort.

toString

```
public static java.lang.String toString(double d)
```

Wandelt eine double-Zahl in einen String um. Bei der üblichen Umwandlung von double-Werten in einen String findet eine Rundung statt. So wird etwa die Zahl 0.1, obwohl intern nicht darstellbar, dennoch auf dem Bildschirm ausgegeben. Diese Methode umgeht die Rundung

[Package](#) [Class](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
