| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-all.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | | |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/BreezySwing/DoubleField.html)   [**NEXT CLASS**](http://docs.google.com/BreezySwing/GBApplet.html) | [**FRAMES**](http://docs.google.com/index.html)    [**NO FRAMES**](http://docs.google.com/Format.html) |
| SUMMARY:  INNER | FIELD | [CONSTR](#1fob9te) | [METHOD](#3znysh7) | DETAIL:  FIELD | [CONSTR](#tyjcwt) | [METHOD](#1t3h5sf) |

## **BreezySwing**

Class Format

java.lang.Object  
 |  
 +--**BreezySwing.Format**

public class **Format**extends java.lang.Object

The class Format contains methods to format data that are left-justified, right-justified, or centered within a given number of columns.

| **Constructor Summary** | |
| --- | --- |
| [**Format**](http://docs.google.com/BreezySwing/Format.html#Format())() |

| **Method Summary** | |
| --- | --- |
| static java.lang.String | [**justify**](http://docs.google.com/BreezySwing/Format.html#justify(char,%20char,%20int))(char leftRight, char ch, int width)            Converts a character to a string and returns it formatted formatted according to the justification type and the specified width. |
| static java.lang.String | [**justify**](http://docs.google.com/BreezySwing/Format.html#justify(char,%20double,%20int,%20int))(char leftRight, double x, int width, int precision)            Converts a double to a string and returns it formatted according to the justification type and the specified width and precision. |
| static java.lang.String | [**justify**](http://docs.google.com/BreezySwing/Format.html#justify(char,%20long,%20int))(char leftRight, long x, int width)            Converts a long to a string and returns it formatted formatted according to the justification type and the specified width. |
| static java.lang.String | [**justify**](http://docs.google.com/BreezySwing/Format.html#justify(char,%20java.lang.String,%20int))(char leftRight, java.lang.String str, int width) |
| static void | [**testJustify**](http://docs.google.com/BreezySwing/Format.html#testJustify())() |

| **Methods inherited from class java.lang.Object** |
| --- |
| clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait |

| **Constructor Detail** |
| --- |

### Format

public **Format**()

| **Method Detail** |
| --- |

### justify

public static java.lang.String **justify**(char leftRight,  
 java.lang.String str,  
 int width)

### justify

public static java.lang.String **justify**(char leftRight,  
 char ch,  
 int width)

Converts a character to a string and returns it formatted formatted according to the justification type and the specified width. **Parameters:**leftRight - the type of justification ('l', 'c', or 'r').ch - the character to be formatted.width - the number of columns in which the character is placed.

### justify

public static java.lang.String **justify**(char leftRight,  
 long x,  
 int width)

Converts a long to a string and returns it formatted formatted according to the justification type and the specified width. **Parameters:**leftRight - the type of justification ('l', 'c', or 'r').x - the long integer to be formatted.width - the number of columns in which the integer is placed.

### justify

public static java.lang.String **justify**(char leftRight,  
 double x,  
 int width,  
 int precision)

Converts a double to a string and returns it formatted according to the justification type and the specified width and precision. The decimal point occupies a column in the formatted number. **Parameters:**leftRight - the type of justification ('l', 'c', or 'r').x - the number to be formatted.width - the number of columns in which the number is placed.precision - the number of places of precision retained in the formatted number. Examples:

String fourPlaces = Format.justify('r', 3.1416, 7, 4);  
 String threePlaces = Format.justify('r', 3.1416, 7, 3);  
 String twoPlaces = Format.justify('r', 3.1416, 7, 2);  
 String noPlaces = Format.justify('r', 3.1416, 7, 0);  
  
 fourPlaces now refers to " 3.1416"  
 threePlaces now refers to " 3.142"  
 twoPlaces now refers to " 3.14"  
 noPlaces now refers to " 3"

### testJustify

public static void **testJustify**()

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-all.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | | |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/BreezySwing/DoubleField.html)   [**NEXT CLASS**](http://docs.google.com/BreezySwing/GBApplet.html) | [**FRAMES**](http://docs.google.com/index.html)    [**NO FRAMES**](http://docs.google.com/Format.html) |
| SUMMARY:  INNER | FIELD | [CONSTR](#1fob9te) | [METHOD](#3znysh7) | DETAIL:  FIELD | [CONSTR](#tyjcwt) | [METHOD](#1t3h5sf) |