

## Source for java.lang.Exception

```
1: /* Exception.java -- generic exception thrown to indicate an exceptional
2:    condition has occurred.
3:    Copyright (C) 1998, 1999, 2001, 2002, 2005  Free Software Foundation, Inc.
4:
5: This file is part of GNU Classpath.
6:
7: GNU Classpath is free software; you can redistribute it and/or modify
8: it under the terms of the GNU General Public License as published by
9: the Free Software Foundation; either version 2, or (at your option)
10: any later version.
11:
12: GNU Classpath is distributed in the hope that it will be useful, but
13: WITHOUT ANY WARRANTY; without even the implied warranty of
14: MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
15: General Public License for more details.
16:
17: You should have received a copy of the GNU General Public License
18: along with GNU Classpath; see the file COPYING.  If not, write to the
19: Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA
20: 02110-1301 USA.
21:
22: Linking this library statically or dynamically with other modules is
23: making a combined work based on this library.  Thus, the terms and
24: conditions of the GNU General Public License cover the whole
25: combination.
26:
27: As a special exception, the copyright holders of this library give you
28: permission to link this library with independent modules to produce an
29: executable, regardless of the license terms of these independent
30: modules, and to copy and distribute the resulting executable under
31: terms of your choice, provided that you also meet, for each linked
32: independent module, the terms and conditions of the license of that
33: module.  An independent module is a module which is not derived from
34: or based on this library.  If you modify this library, you may extend
35: this exception to your version of the library, but you are not
36: obligated to do so.  If you do not wish to do so, delete this
37: exception statement from your version. */
38:
39:
40: package java.lang;
41:
42: /**
43:  * The root class of all exceptions worth catching in a program.  This
44:  * includes the special category of RuntimeException, which
45:  * does not need to be declared in a throws clause.  Exceptions can be used
46:  * to represent almost any exceptional behavior, such as programming errors,
47:  * mouse movements, keyboard clicking, etc.
48:  *
49:  * @author Brian Jones
50:  * @author Warren Levy (warrenl@cygnus.com)
51:  * @author Eric Blake (ebb9@email.byu.edu)
52:  * @status updated to 1.4
53:  */
54: public class Exception extends Throwable
55: {
56:     /**
57:      * Compatible with JDK 1.0+.
```

```

58:  */
59:  private static final long serialVersionUID = -3387516993124229948L;
60:
61:  /**
62:   * Create an exception without a message. The cause remains uninitialized.
63:   *
64:   * @see #initCause(Throwable)
65:   */
66:  public Exception()
67:  {
68:  }
69:
70:  /**
71:   * Create an exception with a message. The cause remains uninitialized.
72:   *
73:   * @param s the message
74:   * @see #initCause(Throwable)
75:   */
76:  public Exception(String s)
77:  {
78:      super(s);
79:  }
80:
81:  /**
82:   * Create an exception with a message and a cause.
83:   *
84:   * @param s the message string
85:   * @param cause the cause of this error
86:   * @since 1.4
87:   */
88:  public Exception(String s, Throwable cause)
89:  {
90:      super(s, cause);
91:  }
92:
93:  /**
94:   * Create an exception with a given cause, and a message of
95:   * <code>cause == null ? null : cause.toString()</code>.
96:   *
97:   * @param cause the cause of this exception
98:   * @since 1.4
99:   */
100: public Exception(Throwable cause)
101: {
102:     super(cause);
103: }
104: }

```

[Overview](#)
[Package](#)
[Class](#)
[Use](#)
[Source](#)
[Tree](#)
[Index](#)
[Deprecated](#)
[About](#)
[GNU Classpath](#) (0.95)