Class java.lang.Long

```
java.lang.Object
   |
   +----java.lang.Number
           |
           +----java.lang.Long
```

---

public final class **Long**
extends Number

The Long class provides an object wrapper for Long data values and serves as a place for long−oriented operations. A wrapper is useful because most of Java's utility classes require the use of objects. Since longs are not objects in Java, they need to be "wrapped" in a Long instance.

Version:
1.20, 10/04/95
Author:
Lee Boynton, Arthur van Hoff

Variable Index

o **MAX_VALUE**
The maximum value a Long can have.
o **MIN_VALUE**
The minimum value a Long can have.

Constructor Index

o **Long**(long)
Constructs a Long object initialized to the specified value.
o **Long**(String)
Constructs a Long object initialized to the value specified by the String parameter.

Method Index

o **doubleValue**()
Returns the value of this Long as a double.
o **equals**(Object)
Compares this object against the specified object.
o **floatValue**()
Returns the value of this Long as a float.
o **getLong**(String)
Get a Long property.
o **getLong**(String, long)
Get a Long property.
o **getLong**(String, Long)
Get a Long property.
o **hashCode**()
Computes a hashcode for this Long.
o **intValue**()
Returns the value of this Long as an int.
o **longValue**()
Returns the value of this Long as a long.
o **parseLong**(String, int)
Assuming the specified String represents a long, returns that long's value.
o **parseLong**(String)
Assuming the specified String represents a long, return that long's value.
o **toString**(long, int)
Returns a new String object representing the specified long in the specified radix.
o **toString**(long)
Returns a new String object representing the specified integer.
o **toString**()
Returns a String object representing this Long's value.
o **valueOf**(String, int)
Assuming the specified String represents a long, returns a new Long object initialized to that value.
o **valueOf**(String)
Assuming the specified String represents a long, returns a new Long object initialized to that value.

Variables

o **MIN_VALUE**

```
public final static long MIN_VALUE
```

The minimum value a Long can have. The lowest minimum value that a Long can have is 0x8000000000000000.

o **MAX_VALUE**

```
public final static long MAX_VALUE
```

The maximum value a Long can have. The larget maximum value that a Long can have is 0x7fffffffffffffff.

Constructors

o **Long**

```
public Long(long value)
```

Constructs a Long object initialized to the specified value.
Parameters:
value – the initial value of the Long

o **Long**

```
public Long(String s) throws NumberFormatException
```

Constructs a Long object initialized to the value specified by the String parameter. The radix is assumed to be 10.
Parameters:
s – the String to be converted to a Long
Throws: NumberFormatException
If the String does not contain a parsable long.

Methods

o **toString**

```
public static String toString(long i,
                              int radix)
```

Returns a new String object representing the specified long in the specified radix.
Parameters:
i – the long to be converted
radix – the radix
See Also:
MIN_RADIX, MAX_RADIX

o **toString**

```
public static String toString(long i)
```

Returns a new String object representing the specified integer. The radix is assumed to be 10.
Parameters:
i – the long to be converted

o **parseLong**

```
public static long parseLong(String s,
                             int radix) throws NumberFormatException
```

Assuming the specified String represents a long, returns that long's value. Throws an exception if the String cannot be parsed as a long.

Parameters:

s – the String containing the integer

radix – the radix to be used

Throws: NumberFormatException

If the String does not contain a parsable integer.

o **parseLong**

```
public static long parseLong(String s) throws NumberFormatException
```

Assuming the specified String represents a long, return that long's value. Throws an exception if the String cannot be parsed as a long. The radix is assumed to be 10.

Parameters:

s – the String containing the long

Throws: NumberFormatException

If the string does not contain a parsable long.

o **valueOf**

```
public static Long valueOf(String s,
                           int radix) throws NumberFormatException
```

Assuming the specified String represents a long, returns a new Long object initialized to that value. Throws an exception if the String cannot be parsed as a long.

Parameters:

s – the String containing the long.

radix – the radix to be used

Throws: NumberFormatException

If the String does not contain a parsable long.

o **valueOf**

```
public static Long valueOf(String s) throws NumberFormatException
```

Assuming the specified String represents a long, returns a new Long object initialized to that value. Throws an exception if the String cannot be parsed as a long. The radix is assumed to be 10.

Parameters:

s – the String containing the long

Throws: NumberFormatException

If the String does not contain a parsable long.

o **intValue**

```
public int intValue()
```

Returns the value of this Long as an int.
Overrides:
intValue in class Number

o **longValue**

```
public long longValue()
```

Returns the value of this Long as a long.
Overrides:
longValue in class Number

o **floatValue**

```
public float floatValue()
```

Returns the value of this Long as a float.
Overrides:
floatValue in class Number

o **doubleValue**

```
public double doubleValue()
```

Returns the value of this Long as a double.
Overrides:
doubleValue in class Number

o **toString**

```
public String toString()
```

Returns a String object representing this Long's value.
Overrides:
toString in class Object

o **hashCode**

```
public int hashCode()
```

Computes a hashcode for this Long.
Overrides:
hashCode in class Object

o **equals**

```
public boolean equals(Object obj)
```

Compares this object against the specified object.

Parameters:

obj – the object to compare with

Returns:

true if the objects are the same; false otherwise.

Overrides:

equals in class Object

o **getLong**

```
public static Long getLong(String nm)
```

Get a Long property. If the property does not exist, it will return 0.

Parameters:

nm – the property name

o **getLong**

```
public static Long getLong(String nm,
                           long val)
```

Get a Long property. If the property does not exist, it will return val. Deals with Hexadecimal and octal numbers.

Parameters:

nm – the String name

val – the Long value

o **getLong**

```
public static Long getLong(String nm,
                           Long val)
```

Get a Long property. If the property does not exist, it will return val. Deals with Hexadecimal and octal numbers.

Parameters:

nm – the property name

val – the Long value